NONLINEAR SURFACE APPROXIMATION

USING PHOTOGAMMETRY

A Thesis

by

ELIZABETH OSGOOD

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2005

Major Subject: Aerospace Engineering

NONLINEAR SURFACE APPROXIMATION

USING PHOTOGAMMETRY

A Thesis

by

ELIZABETH OSGOOD

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	John L. Junkins
Committee Members,	John Hurtado
	Alan Palazzolo
Head of Department,	Helen Reed

December 2005

Major Subject: Aerospace Engineering

ABSTRACT

Nonlinear Surface Approximation
Using Photogammetry. (December 2005)
Elizabeth Osgood, B.S., Embry-Riddle Aeronautical University
Chair of Advisory Committee: Dr. John L. Junkins

Many satellite applications require a model that represents a surface as it deforms over time. Yet, space applications demand a precise, low-weight, low-volume, and easy to implement solution. A metrology sensing system is presented in this thesis, consisting of a series of cameras and laser dot projectors positioned along the length of the antenna. This system accurately models the geometry of the surface to meet the demands of a space based radar. Each laser dot projector casts a matrix of points onto the antenna surface. The points are then imaged simultaneously by a pair of cameras, each having a different, but overlapping, viewpoint. Given the two overlapping images, a Gaussian nonlinear least squares algorithm solves the stereo-triangulation problem which provides the coordinates of the projected points and thereby maps the surface.

There are three different strategies discussed in this thesis. The first strategy assumes the positions and orientations of the cameras are absolutely known. This produces an extremely accurate result; yet it is unrealistic to assume absolute knowledge of cameras locations and orientations for the application. The next strategy assumes the positions and orientations of the cameras are completely unknown in addition to the unknown surface. This program produces a less accurate, but more realistic, result considering the dynamic nature of rigid structures in space. To increase the accuracy and improve the robustness of these results, the third method employs a global metrology sensing system to reduce the uncertainty in the location and orientation of the outboard cameras relative to the center camera. This approach estimates the surface extremely

accurately and, although more complex, offers advantages and addresses the desire for a family of designs wherein higher accuracy is achievable by further optimization.

ACKNOWLEDGEMENTS

I would like to thank Dr. John L. Junkins for his constant support and guidance during my studies at Texas A&M. I would also like to thanks my professors, Dr. John Hurtado for his great understanding and academic assistance, and Dr. Daniele Mortari for his persistence in searching for a linearization method. Additionally, I would like to thank Puneet Singla and Stefanie Beaver for their aid during the programming process; they provided essential computational support. Finally, I would like to thank my family, who provide me with daily support, encouragement, love, and comic relief just when I need it.

TABLE OF CONTENTS

Page

Page

LIST OF FIGURES

Page

Page

Page

CHAPTER I

INTRODUCTION

A simple and accurate method for modeling the surface geometry of an antenna surface relative to a global reference frame is needed for many space applications. Considering the restrictions of space, if the measurement system can be low weight, small, and low cost, it is obviously more desirable. Also, the system can be used in a variety of alternative ways including mapping the deformation of satellite subsystems. This thesis will address the modeling of a space-based radar (SBR) satellite antenna using a system comprised of cameras and laser dot projectors. A pair of cameras image the SBR antenna surface, using a matrix of dots (called points) produced by the laser dot projector. With the advance in imaging technology, these items are low cost, small, and as implemented in this study, can produce accurate results.

There are many modeling systems in existence that use structured light as their modeling tool.[1] There are also systems that use stereo triangulation to image and model a surface.[2] This thesis explores the use of triangulation to measure the positions of points projected using a laser dot projector (structured light) processed via a Gaussian least squares algorithm. The Gaussian algorithm is commonly used for spacecraft navigation,[3] but this thesis will adapt the combination of sensing and estimation algorithm to accurately model an SBR antenna surface.

In Chapter II, the colinearity equations are derived for a single camera. In Chapter III, stereo-triangulation ideas are explained for a stereo pair of cameras, called the *Local Environment*, utilizing the colinearity equations. The Gaussian nonlinear least squares estimation technique is then applied to a stereo-pair. Chapter IV contains the composition of the stereo-pairs to an entire surface with multiple cameras, called the *Global Environment*.

This thesis follows the style of the *Journal of Guidance, Control, and Dynamics.*

In addition to the theory, Chapters III and IV apply the methods developed in those chapters to the following three strategies for modeling the surface geometry of the SBR antenna surface relative to a global reference frame:

i. known camera position and attitude: only the locations on the surface to be modeled are unknown,

ii. unknown camera position and attitude: the locations on the surface and location(s) and attitude(s) of the camera(s) are unknown, and

iii. measured camera position and attitude: the locations on the surface are unknown and the location(s) and attitude(s) of the camera(s) are measured using sensors such as those described in Appendix B. More detail on these sensors will be introduced in Chapter III.

The thesis then concludes with a summary and the conclusions drawn from the previous chapters.

CHAPTER II

SINGLE CAMERA THEORY

Colinearity Equations

The most basic projection equations that map object geometry into image coordinates is considered in this chapter. This derivation is for a single pin-hole model camera. The next chapter applies this derivation to a pair of cameras. Figure 1 shows the configuration of the surface to be modeled, the camera lens, and the negative image plane. The distance between the negative image plane and the lens is defined as the focal length, which is also equal to the distance between the positive image and the lens. The positive image is a transformation of the negative image through the lens.

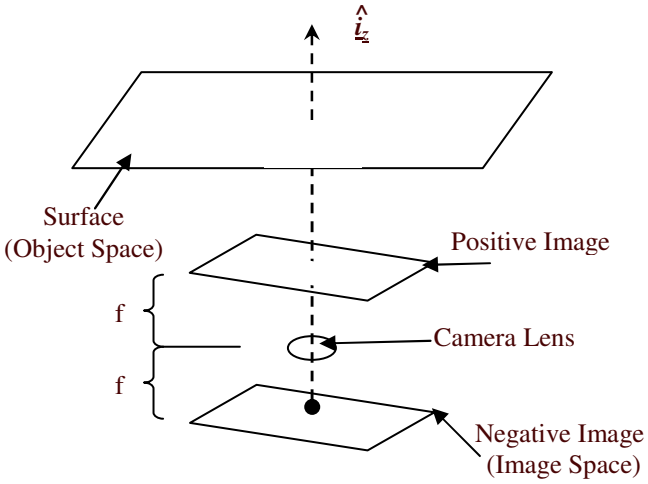


Fig. 1 Pin-Hole Camera.

To develop the model, two co-ordinate frames must be defined. The first frame is the global reference frame (Figure 2), also called the *object space*. The coordinates (X_c, Y_c, and Z_c) locate the principal point on the lens. The location on the surface has

coordinates (*X, Y,* and *Z*). The location imaged onto the negative image has coordinates ($[x]_g$, $[y]_g$, $-f$) where the subscript *g* denotes the global coordinate system.

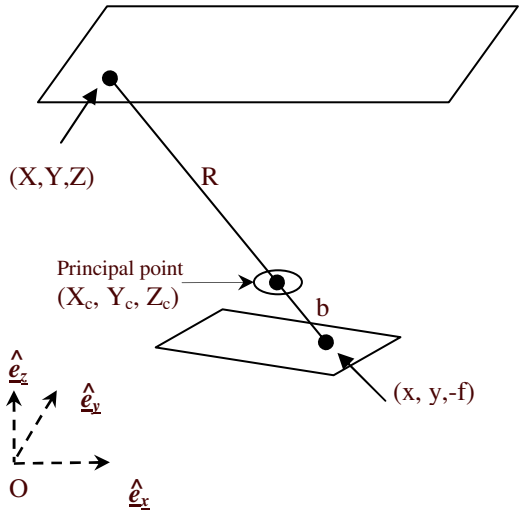


Fig. 2 Global (Object Space) Reference Frame.

The second frame is the camera reference frame (Figure 3), also called the *image space*. The origin of the image space is at (x_o, y_o, $-f$), where f represents the fixed focal length from the lens of the camera to the image, as shown from Figure 1. The point on the negative image can also be described in image space, with the notation ($[x]_c$, $[y]_c$, $-f$), where *c* represents the camera reference frame.

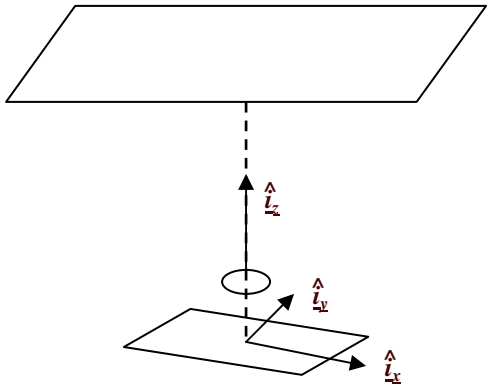


Fig. 3 Camera (Image Space) Reference Frame.

The direction cosine matrix C prescribes the rotational displacement of the camera reference frame relative to the global reference frame. These reference frames can now be used to describe the variables employed in the colinearity equations. In global space, the position of a point on the surface is related to its location on the camera negative. The vector from the point on the antennae to the lens, (X,Y,Z) to (X_c,Y_c,Z_c), is defined as $\underline{R}$, and the vector from the point on the negative to the lens, $(x,y,-f)$ to $(x_o, y_o, 0)$, is defined as $\underline{b}$, as shown in Figure 2. The scalar form, or magnitude of $\underline{R}$ is referred to as R and the magnitude of $\underline{b}$ is referred to as b.

Object space components:

$$\underline{R} \equiv \begin{pmatrix} X - X_c \\ Y - Y_c \\ Z - Z_c \end{pmatrix} \tag{1}$$

$$R = \sqrt{(X - X_c)^2 + (Y - Y_c)^2 + (Z - Z_c)^2} \tag{2}$$

Image space components:

$$\underline{b} \equiv \begin{pmatrix} x - x_o \\ y - y_o \\ -f \end{pmatrix} \tag{3}$$

$$b = \sqrt{(x - x_o)^2 + (y - y_o)^2 + f^2} \tag{4}$$

According to the colinearity idealization associated with the pinhole camera model, the two vectors ($\underline{R}$ and $\underline{b}$) are colinear and form similar triangles with respect to the Z-axis. The norm of the vector is preserved, so the colinearity relation between $\underline{R}$ and $\underline{b}$ is then easily established.

$$\underline{R} = \alpha \underline{b} \tag{5}$$

This equation holds for abstract vectors ($\underline{R}$, $\underline{b}$). To determine the value of α, the scalar form of R and b is used.

$$\alpha = \frac{|\underline{R}|}{|\underline{b}|} = \sqrt{\frac{(X - X_C)^2 + (Y - Y_C)^2 + (Z - Z_C)^2}{(x - x_o)^2 + (y - y_o)^2 + f^2}} = \sqrt{\frac{\underline{R} \cdot \underline{R}}{\underline{b} \cdot \underline{b}}} \tag{6}$$

When the corresponding component or matrix equation is written, the components must be projected to a common reference frame, using the orthogonal direction cosine matrix *C*. The relation between the set of components of Eq. (5), with $\underline{R}$ components in object space and $\underline{b}$ components in image space gives:

$$\begin{pmatrix} X - X_c \\ Y - Y_c \\ Z - Z_c \end{pmatrix} = \alpha C^T \begin{pmatrix} x - x_o \\ y - y_o \\ -f \end{pmatrix} \tag{7}$$

Eq. (6) also follows from Eq (7), since the magnitude of a vector is unchanged by an orthogonal projection. Solving for the image space components from Eq. (7) produces the vector form of the colinearity equations.

$$\begin{pmatrix} x - x_o \\ y - y_o \\ -f \end{pmatrix} = \alpha[C] \begin{pmatrix} X - X_c \\ Y - Y_c \\ Z - Z_c \end{pmatrix} \tag{8}$$

In this thesis, the focal length is assumed to be fixed. Note this equation can be solved for x and y, which projects an object space point (X, Y, Z) into (x, y), the corresponding image point.

$$x = x_0 - \frac{1}{\alpha}(C_{11}(X - X_C) + C_{12}(Y - Y_C) + C_{13}(Z - Z_C)) \tag{9}$$

$$y = y_0 - \frac{1}{\alpha}(C_{21}(X - X_C) + C_{22}(Y - Y_C) + C_{23}(Z - Z_C)) \tag{10}$$

$$0 = f - \frac{1}{\alpha}(C_{31}(X - X_C) + C_{32}(Y - Y_C) + C_{33}(Z - Z_C)) \tag{11}$$

Eq. (11) is rearranged to solve for α.

$$\alpha = -\frac{1}{f}(C_{13}(X - X_C) + C_{23}(Y - Y_C) + C_{33}(Z - Z_C)) \tag{12}$$

Finally, this is inserted into Eq. (9) and (10), which transforms three equations into two, to give a familiar form for the colinearity equations, which map points from object space to image space.

$$x = x_0 - f\frac{(C_{11}(X - X_C) + C_{12}(Y - Y_C) + C_{13}(Z - Z_C))}{(C_{31}(X - X_C) + C_{32}(Y - Y_C) + C_{33}(Z - Z_C))} \tag{13}$$

$$y = y_0 - f\frac{(C_{21}(X - X_C) + C_{22}(Y - Y_C) + C_{23}(Z - Z_C))}{(C_{31}(X - X_C) + C_{32}(Y - Y_C) + C_{33}(Z - Z_C))} \tag{14}$$

Attitude Parameterization

The rotation matrix for each camera is parameterized using Modified Rodrigues Parameters (MRP).[4] While there are many attitude parameterizations that can be used, MRPs are chosen because a singular result is easily avoided. The MRPs vector consists of three coordinates.

$$\underline{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{15}$$

The MRP vector described is in terms of the principle rotation, Φ, which can have a value of up to 360° before a singularity is encountered.

$$\underline{q} = \tan\frac{\Phi}{4}\underline{\hat{e}} \tag{16}$$

The direction cosine matrix for MRP is described in vector form, using a skew-symmetric matrix. In index notation form, MRPs are described as q_k, where $1 \leq k \leq 3$.

$$[C] = [I_{3x3}] + \frac{8[\underline{\tilde{q}}]^2 - 4(1 - \underline{q}^T\underline{q})[\underline{\tilde{q}}]}{(1 + \underline{q}^T\underline{q})^2} \tag{17}$$

$$\left[\tilde{\underline{q}}\right] = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \tag{18}$$

A point's location can now be projected from the surface of the antenna to its location in the image frame. The next chapter discusses how to reverse the transformation to find the point's location on the antenna from the image plane.

CHAPTER III

LOCAL ENVIRONMENT

Stereo Triangulation

Stereo-triangulation employs a pair of cameras and a laser dot projector (Figure 4). As previously mentioned, the dots projected onto the surface are called *points*. The projected points must be sufficiently dense to adequately define the surface geometry. The origin of the global reference frame (Figure 5) is located at the lens of camera 1. The origin of the camera reference frame is located at the lens of camera 2, so the two images have separate reference frames. Both cameras are assumed to have a 90° field of view.

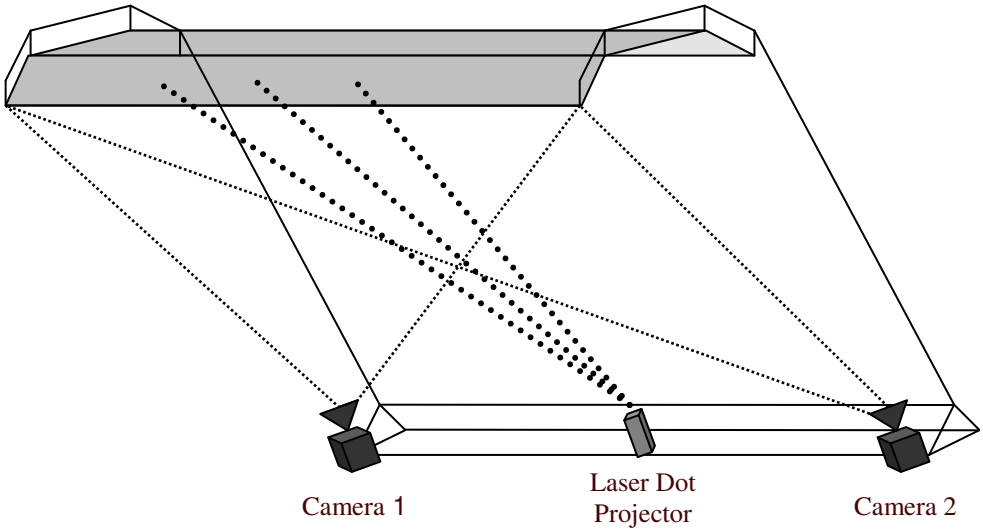


Fig. 4 Local Sensing.

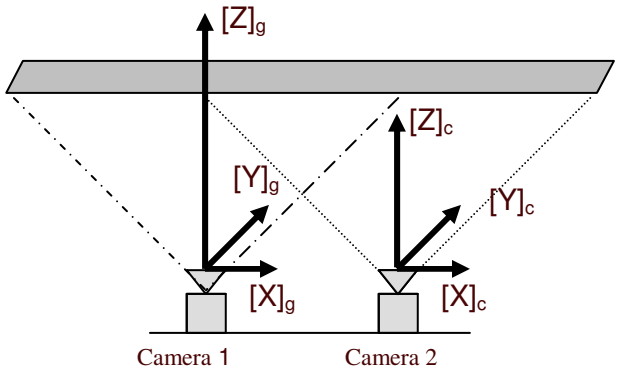


Fig. 5 Stereo-Pair Coordinate Frames.

To apply the colinearity equations to a pair of cameras, Eq. (9) and (10) must first be put in a more general form which includes j=[1,2] cameras and i=[1,2,··· p] total points on the surface to image.

$$x_{ij} = x_{0j} - f\frac{(C_{11j}(X_i - X_C) + C_{12j}(Y_i - Y_C) + C_{13j}(Z_i - Z_C))}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{19}$$

$$y_{ij} = y_{0j} - f\frac{(C_{21j}(X_i - X_{Cj}) + C_{22j}(Y_i - Y_{Cj}) + C_{23j}(Z_i - Z_{Cj}))}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{20}$$

There are two variables (x_{ij}, y_{ij}), but each point has three coordinates in three-dimensional space (X_i, Y_i, Z_i). This is an underdetermined system, with only two equations for three unknown quantities. To solve the problem, there must more equations than unknown quantities; two different images are required, with a sufficient number of total measured points to determine all unknowns. This approach is called stereo-triangulation.

The image produced by camera 1 (Figure 6) will not be identical to the image from camera 2 (Figure 7). The portion that overlaps on the two images (Figure. 8) defines the surface to be imaged. [5]

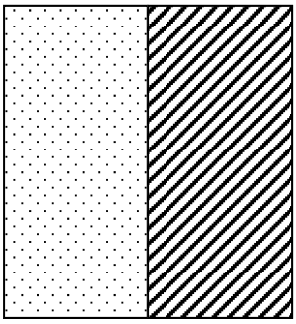


Fig. 6 Image from Camera 1.

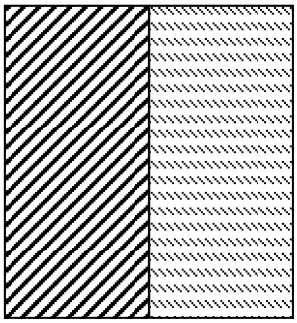


Fig. 7 Image from Camera 2.

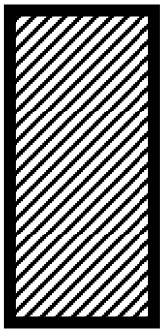


Fig. 8 Modeled Portion.

Gaussian Nonlinear Least Squares

Due to the algebraic complexity and nonlinearity of the colinearity equations, it is not feasible to analytically solve for each variable (X_i, Y_i, and Z_i). Instead, these coupled nonlinear equations can be solved using Gauss's nonlinear least squares solution.[6] The following derivation is specialized to solve the problem where only X_i, Y_i, and Z_i are unknown.

Gauss's least squares differential algorithm is used to minimize the sum square of the residual errors. The notation adopted for any variable is: true ($\underline{x}$), measured ($\underline{\tilde{x}}$), and estimated ($\underline{\hat{x}}$). The true value has no error, the measured value is determined by the sensors with a measurement error, and the estimated value is iterated and refined to minimize the selected optimality criteria (weighted least square sum of residuals). Perfect measurements are unrealistic and only expected in an ideal situation.

Note the measurement model

$$\underline{\tilde{y}} = f(\underline{x}) + \underline{v} \tag{21}$$

$\underline{\tilde{y}}$ = measured values = $[x_{ij}, y_{ij}]^T$

$\underline{x}$ = true unknown values = $[X_i, Y_i, Z_i]^T$

$f(\underline{x})$ = colinearity equations of unknown values

$\underline{v}$ = measurement error

The parameter estimate ($\underline{\hat{x}}$) is mapped into an estimate ($\underline{\hat{y}}$) of the measurements as

$$\underline{\hat{y}} = f(\underline{\hat{x}}) \tag{22}$$

$\underline{\hat{y}}$ = estimated values = $[\ \underline{\hat{x}_{ij}}, \underline{\hat{y}_{ij}},]^T$

$\underline{\hat{x}}$ = estimated unknown values = $[X_i, Y_i, Z_i]^T$

$f(\hat{x})$ = colinearity equations of estimated values

There are two types of error: measurement error ($\underline{v}$) and residual error ($\underline{e}$). The residual error is the difference between the measured and estimated values of $\underline{y}$. This relationship is used to create an equation relating the estimated and measured values.

$$\underline{e} = \underline{\tilde{y}} - \underline{\hat{y}} \equiv \Delta \underline{y} \tag{23}$$

$$\underline{\tilde{y}} = f(\underline{\hat{x}}) + \underline{e} \tag{24}$$

As previously stated, Gauss's principle of least squares is to minimize the sum square of the residual errors, which uses a weighting matrix W to inform least squares optimization of the validity of each measurement. The weight matrix is composed of diagonal elements of the uncertainty of each measurement squared.

$$J = \frac{1}{2}\underline{e}^T W \underline{e} = \frac{1}{2}\Delta \underline{y}^T W \Delta \underline{y} \tag{25}$$

$$J = \frac{1}{2}[\underline{\tilde{y}} - f(\underline{\hat{x}})]^T W [\underline{\tilde{y}} - f(\underline{\hat{x}})] \tag{26}$$

Since the colinearity equations are nonlinear, J cannot immediately be minimized as a closed-form solution. Thus, the process is iterated using local linearizations of $f(\underline{x})$, and the equations are modified to include a new variable that contains the unknown values at the current iteration: $\underline{x_C}$.

$$\underline{\hat{x}} = \underline{x}_C + \Delta \underline{x} \tag{27}$$

$\underline{x_C}$ = current estimated unknown values during iteration

$\Delta \underline{x}$ = corrections to the estimated unknown values

The equation that relates $f(\underline{\hat{x}})$ to $f(\underline{x}_C)$ can be linearized by a first order Taylor series expansion, if the correction $\Delta \underline{x}$ is assumed to be small, as

$$f(\underline{\hat{x}}) \approx f(\underline{x}_C) + \left(\frac{\partial f}{\partial \underline{x}}\bigg|_{\underline{x}_C}\right)\Delta \underline{x} \tag{28}$$

The differential of the colinearity equations with respect to the true values, computed at the current estimated value of the unknowns is called the H-matrix or Jacobian matrix.

$$H \equiv \left.\frac{\partial f}{\partial \underline{x}}\right|_{\underline{x}_C} \tag{29}$$

The residual error at each $\underline{x}_C$ is defined as $\Delta \underline{y}_C$, modeled after Eq. (23).

$$\Delta \underline{y}_C \equiv \underline{\tilde{y}} - f(\underline{x}_C) \tag{30}$$

Eq. (27) and (30) are substituted into Eq. (23), which yields the iterated form of $\Delta \underline{y}$. The sum square of the predicted residuals, J_P, is derived by substituting Eq. (30) into Eq. (25).

$$\Delta \underline{y} = \Delta \underline{y}_C - H\Delta \underline{x} \tag{31}$$

$$J_P \equiv \frac{1}{2}(\Delta \underline{y}_C - H\Delta \underline{x})^T W (\Delta \underline{y}_C - H\Delta \underline{x}) \tag{32}$$

J_P is the value to be minimized to determine the estimated value of the unknown $\underline{\hat{x}}$ values. There are two conditions that must be satisfied to minimize J_P. The first derivative with respect to each unknown value must vanish, but this necessary condition is not sufficient to ensure the stationary points are a minimum and not a maximum. The sufficient condition is that the second derivative matrix of J_P with respect to each unknown value must be positive definite to ensure the stationary point is a minimum value of J_P.

$$J_P = \frac{1}{2}(\Delta \underline{y}_C{}^T W \Delta \underline{y}_C - 2\Delta \underline{x}^T H^T W \Delta \underline{y}_C + \Delta \underline{x}^T H^T W H \Delta \underline{x}) \tag{33}$$

$$\nabla_{\Delta \underline{x}} J_P = (-H^T W \Delta \underline{y}_C + H^T W H \Delta \underline{x}) = 0 \tag{34}$$

$$\nabla^2{}_{\Delta \underline{x}} J_P = (H^T W H) > 0 \tag{35}$$

The necessary condition is solved for $\Delta\underline{x}$, giving *Gauss' Normal Equations*.

$$\Delta\underline{x} = (H^T W H)^{-1} H^T W \Delta\underline{y}_C \tag{36}$$

The sufficient condition, namely that $(H^T WH)$ be positive definite for all instances, is satisfied because W contains all positive numbers, and $(H^T WH)$ always has positive eigenvalues if H is of maximum rank (the number of elements in $\underline{x}$). Also, there are always more equations than unknowns, because stereo-triangulation is employed, so the sufficient condition is always positive definite for this application.

The nonlinear least squares approximation technique is summarized in the following flowchart (Figure 9). The process is iterated until the linear approximation is less then some predetermined small number ε.

$$\frac{J(\underline{x}_C + \Delta\underline{x}_C) - J(\underline{x}_C)}{J(\underline{x}_C)} < \varepsilon \tag{37}$$

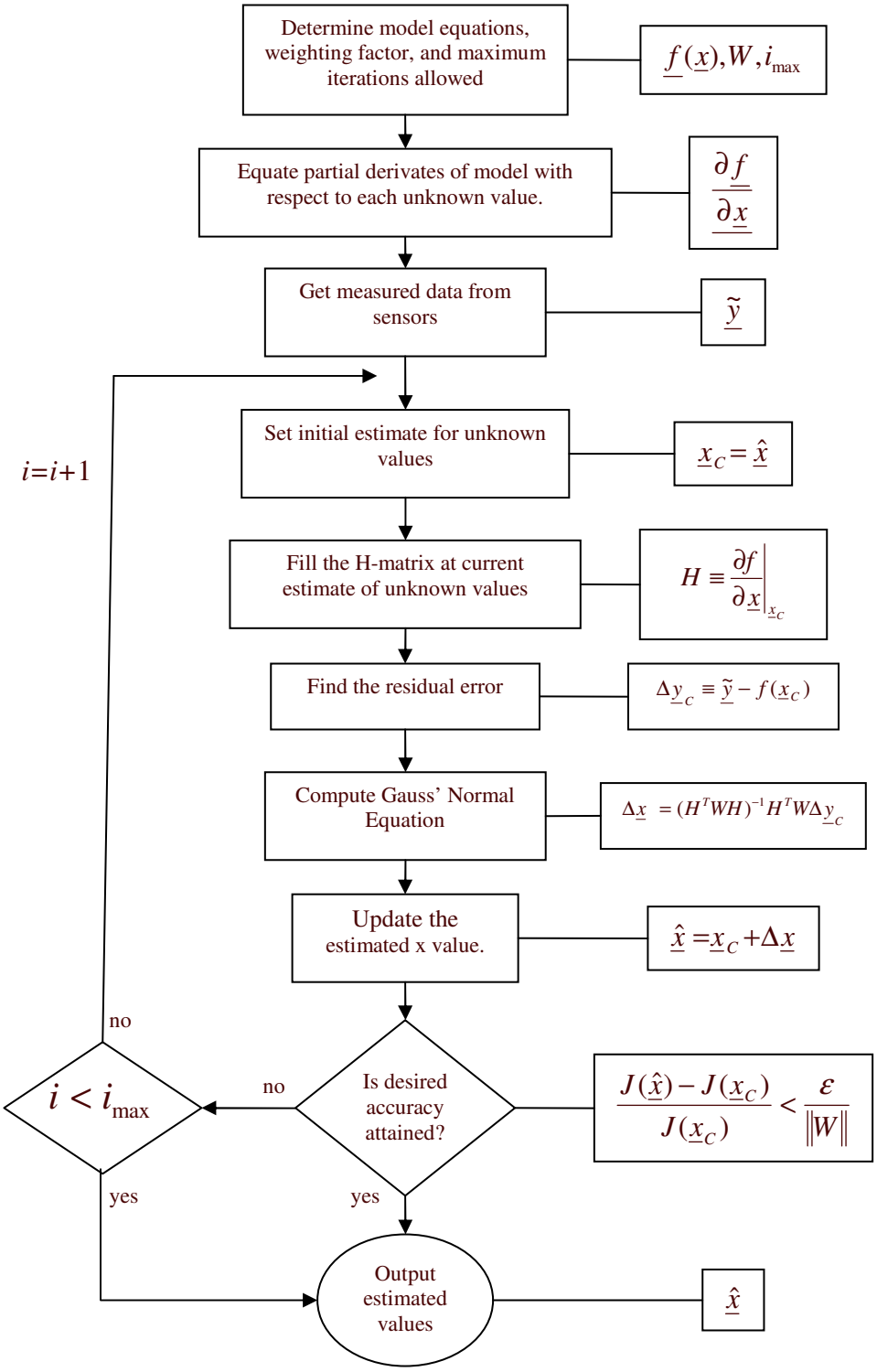


Fig. 9 Gaussian Nonlinear Least Squares Flowchart.

Jacobian Matrix

The *H*-matrix used in the Gaussian nonlinear least squares derivation is the Jacobian matrix of partial derivates of the colinearity equations (x_{ij} and y_{ij}) with respect to each unknown variable. For each of the three strategies discussed in the first chapter of this thesis, there are a different number of unknowns, so each matrix will have a different dimension.

Absolute Camera Position and Attitude

The first strategy assumes the camera position and attitude are known values, called *control points,* and the only unknown values are the point locations, referred to as *pass points*. The colinearity equations are functions containing the following variables of pass and control points:

$$\begin{aligned} x_{ij} &= F(X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, q_{j1}, q_{j2}, q_{j3}, f_j) \\ y_{ij} &= G(X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, q_{j1}, q_{j2}, q_{j3}, f_j) \\ & known: X_{Cj}, Y_{Cj}, Z_{Cj}, q_{j1}, q_{j2}, q_{j3}, f_j \\ & unknown: X_i, Y_i, Z_i \end{aligned} \tag{38}$$

The Jacobian is composed of the partial derivative of the colinearity equations with respect to each unknown value.

$$H = \begin{bmatrix} \frac{\partial x_{ij}}{\partial X_i}, & \frac{\partial x_{ij}}{\partial Y_i}, & \frac{\partial x_{ij}}{\partial Z_i} \\ \frac{\partial y_{ij}}{\partial X_i}, & \frac{\partial y_{ij}}{\partial Y_i}, & \frac{\partial y_{ij}}{\partial Z_i} \end{bmatrix} \tag{39}$$

Due to the nature of the colinearity equations, the partial of one point with respect to another is given a zero value in the *H*-matrix. The following matrix shows a populated portion of the *H*-matrix.

In this first application, the *H*-matrix contains 4*p rows and 3*p columns, which gives the *H*-matrix a dimension of (24x18) for a surface with 6 points. Figure 10 shows the locations of nonzero values in a full *H*-matrix. The darker color shows the partial derivatives for camera 1 and the lighter color shows the partial derivatives for camera 2. If a cell has no color, is has a value of 0.

$$\frac{\partial x_{ij}}{\partial X_i} = \begin{bmatrix} \frac{\partial x_{11}}{\partial X_1} & 0 & \cdots & 0 & \cdots & 0 \\ \frac{\partial x_{12}}{\partial X_1} & 0 & \cdots & 0 & \cdots & 0 \\ 0 & \frac{\partial x_{21}}{\partial X_2} & \cdots & 0 & \cdots & 0 \\ 0 & \frac{\partial x_{22}}{\partial X_2} & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & \cdots & \frac{\partial x_{i1}}{\partial X_i} & \cdots & 0 \\ 0 & 0 & \cdots & \frac{\partial x_{i2}}{\partial X_i} & \cdots & 0 \\ \vdots & \vdots & & & & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & \frac{\partial x_{p1}}{\partial X_p} \\ 0 & 0 & \cdots & 0 & \cdots & \frac{\partial x_{p2}}{\partial X_p} \end{bmatrix} \tag{40}$$

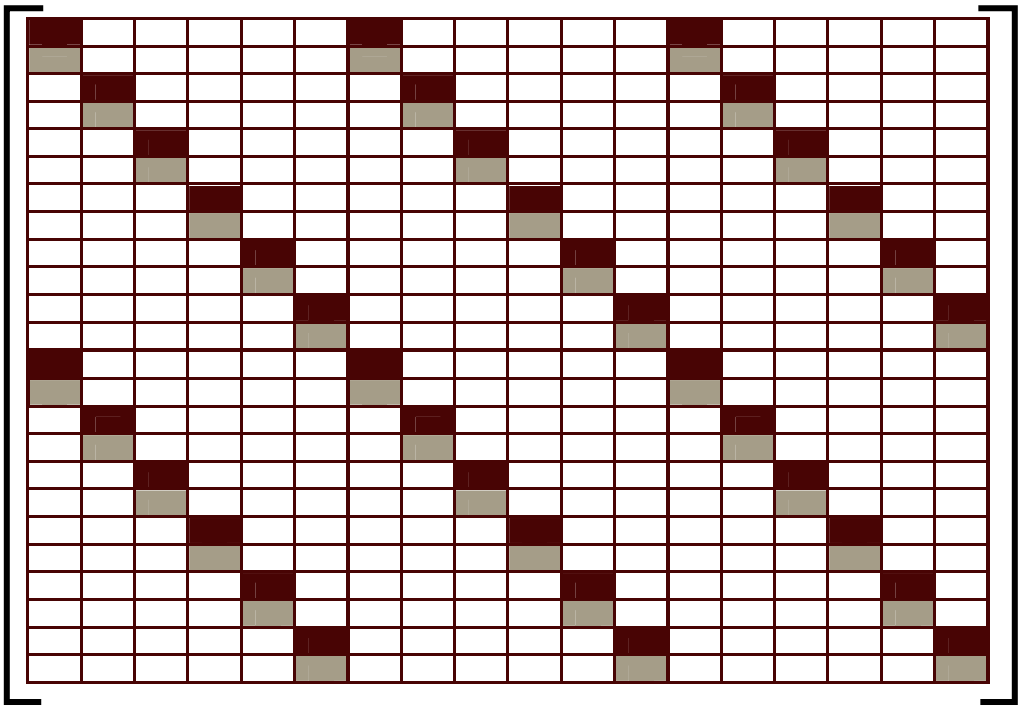


Fig. 10 Populated *H*-Matrix for Strategy 1.

The more points there are, the more accurate the surface is represented, because the measured points are denser. However, this increase in the number of unknown values increases the number of computations in the algorithm, which increases the run time.

The following equations are the partial derivatives of the colinearity equations for each unknown value. To simplify the equations and thereby reduce the number of computations, the values for x_{ij} and y_{ij}, which are computed earlier in the algorithm, are inserted.

$$\frac{\partial(x_{ij})}{\partial X_i} = \frac{(-f * C_{11j} - C_{31j} * x_{ij})}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{41}$$

$$\frac{\partial(x_{ij})}{\partial Y_i} = \frac{(-f * C_{12j} - C_{32j} * x_{ij})}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{42}$$

$$\frac{\partial(x_{ij})}{\partial Z_i} = \frac{(-f * C_{13j} - C_{33j} * x_{ij})}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{43}$$

$$\frac{\partial(y_{ij})}{\partial X_i} = \frac{(-f * C_{21} - C_{31} * y_{ij})}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{44}$$

$$\frac{\partial(y_{ij})}{\partial Y_i} = \frac{(-f * C_{22j} - C_{32j} * y_{ij})}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{45}$$

$$\frac{\partial(y_{ij})}{\partial Z_i} = \frac{(-f * C_{23j} - C_{33j} * y_{ij})}{(C_{31j}(X_i - X_{Cj}) + C_{32j}(Y_i - Y_{Cj}) + C_{33j}(Z_i - Z_{Cj}))} \tag{46}$$

Since the *H*-matrix contains the derivative of the equations being solved, a simple method to validate the *H*-matrix values is to use the following equation with a very small number $\varepsilon \leq 10^{-5}$ m for each unknown value. This approximation to validate the *H*-matrix is accurate to 10^{-5} m.

$$\frac{df}{dX} \approx \frac{1}{\varepsilon}(f(X + \varepsilon) - f(X)) \tag{47}$$

$$\frac{d([x_{ij}, \quad y_{ij}])}{dX_i} \approx \frac{1}{\varepsilon}(f(X_i + \varepsilon, \quad Y_i, \quad Z_i) - f(X_i, \quad Y_i, \quad Z_i)) \tag{48}$$

$$\frac{d([x_{ij}, \quad y_{ij}])}{dY_i} \approx \frac{1}{\varepsilon}(f(X_i, \quad Y_i + \varepsilon, \quad Z_i) - f(X_i, \quad Y_i, \quad Z_i)) \tag{49}$$

$$\frac{d([x_{ij}, \quad y_{ij}])}{dZ_i} \approx \frac{1}{\varepsilon}(f(X_i, \quad Y_i, \quad Z_i + \varepsilon) - f(X_i, \quad Y_i, \quad Z_i)) \tag{50}$$

Unknown Camera Position and Attitude

In the second application, the locations and attitudes of the cameras are also being determined. More points are required because while the number of unknown values increases, the number of equations remains the same. Given that the centermost camera is the origin of global space, its location and attitude are known; it is located at (0,0,0) m with an attitude of $q=(0,0,0)^T$. This implies that only 6 additional unknowns are added: the location and attitude of the second camera. The variables in the colinearity equations are categorized as follows:

$$known: X_{C1}, Y_{C1}, Z_{C1}, q_{11}, q_{12}, q_{13}, f_j$$
$$unknown: X_i, Y_i, Z_i, X_{C2}, Y_{C2}, Z_{C2}, q_{21}, q_{22}, q_{23}$$

The *H*-matrix now includes 6 new columns, making the new dimension 4*p rows by 3*p+6 columns. Simple algebraic manipulation of this dimension dictates that there must be at least 6 points to avoid an underdetermined system.

$$H = \begin{bmatrix} \frac{\partial x_{ij}}{\partial X_i}, & \frac{\partial x_{ij}}{\partial Y_i}, & \frac{\partial x_{ij}}{\partial Z_i}, & \frac{\partial x_{ij}}{\partial X_{C2}}, & \frac{\partial x_{ij}}{\partial Y_{C2}}, & \frac{\partial x_{ij}}{\partial Z_{C2}}, & \frac{\partial x_{ij}}{\partial q_{2k}} \\ \frac{\partial y_{ij}}{\partial X_i}, & \frac{\partial y_{ij}}{\partial Y_i}, & \frac{\partial y_{ij}}{\partial Z_i}, & \frac{\partial y_{ij}}{\partial X_{C2}}, & \frac{\partial y_{ij}}{\partial Y_{C2}}, & \frac{\partial y_{ij}}{\partial Z_{C2}}, & \frac{\partial y_{ij}}{\partial q_{2k}} \end{bmatrix} \quad (51)$$

In addition to the previous equations, the partial derivatives for x_{i2} and y_{i2} with respect to X_{c2}, Y_{c2}, Z_{c2}, and q_{k2} are required. The partial derivatives of x_{i1} and y_{i1} with respect to X_{c2}, Y_{c2}, Z_{c2}, and q_{k2} are 0 because the colinearity equations do not relate one camera to another. Figure 11 contains a populated *H*-matrix for 6 points (24 x 24), where the dark value represents a partial derivative with respect to camera 1, and the light value shows a partial derivative with respect to camera 2. The blank cells contain a value of 0.

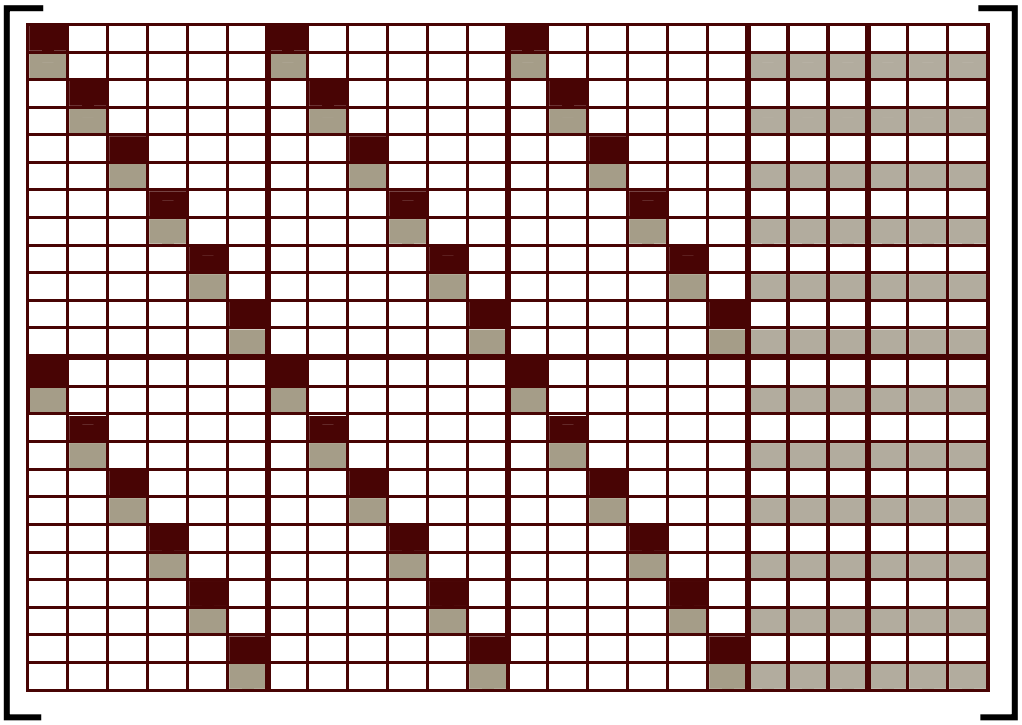


Fig. 11 Populated *H*-Matrix for Strategy 2.

This example contains the minimum number of points. The following partial derivative equations are specific for camera 2, because the position and attitude of camera 1 is known, so partial derivates are not required for camera 1. Taking advantage of the similar relationship between some of the partial derivative equations simplifies the computations.

$$\frac{\partial(x_{i2})}{\partial X_{c2}} = -\frac{\partial(x_{i2})}{\partial X_i} \tag{52}$$

$$\frac{\partial(x_{i2})}{\partial Y_{c2}} = -\frac{\partial(x_{i2})}{\partial Y_i} \tag{53}$$

$$\frac{\partial(x_{i2})}{\partial Z_{c2}} = -\frac{\partial(x_{i2})}{\partial Z_i} \tag{54}$$

$$\frac{\partial(y_{i2})}{\partial X_{c2}} = -\frac{\partial(y_{i2})}{\partial X_i} \tag{55}$$

$$\frac{\partial(y_{i2})}{\partial Y_{c2}} = -\frac{\partial(y_{i2})}{\partial Y_i} \tag{56}$$

$$\frac{\partial(y_{i2})}{\partial Z_{c2}} = -\frac{\partial(y_{i2})}{\partial Z_i} \tag{57}$$

The partial derivatives of the colinearity equations with respect to the MRP vectors are much more complex, so a new variable β is introduced.

$$\frac{\partial(x_{i2})}{\partial q_{k2}} = \frac{1}{\beta}\begin{bmatrix}(X_i - X_{c2})\\(Y_i - Y_{c2})\\(Z_i - Z_{c2})\end{bmatrix}^T \begin{bmatrix}(x_{i2} - x_{o2})\dfrac{\partial(C_{312})}{\partial q_{k2}} - f\dfrac{\partial(C_{112})}{\partial q_{k2}}\\(x_{i2} - x_{o2})\dfrac{\partial(C_{322})}{\partial q_{k2}} - f\dfrac{\partial(C_{122})}{\partial q_{k2}}\\(x_{i2} - x_{o2})\dfrac{\partial(C_{332})}{\partial q_{k2}} - f\dfrac{\partial(C_{132})}{\partial q_{k2}}\end{bmatrix} \tag{58}$$

$$\frac{\partial(y_{i2})}{\partial q_{k2}} = \frac{1}{\beta}\begin{bmatrix}(X_i - X_{c2})\\(Y_i - Y_{c2})\\(Z_i - Z_{c2})\end{bmatrix}^T \begin{bmatrix}(y_{i2} - y_{o2})\dfrac{\partial(C_{312})}{\partial q_{k2}} - f\dfrac{\partial(C_{212})}{\partial q_{k2}}\\(y_{i2} - y_{o2})\dfrac{\partial(C_{322})}{\partial q_{k2}} - f\dfrac{\partial(C_{222})}{\partial q_{k2}}\\(y_{i2} - y_{o2})\dfrac{\partial(C_{332})}{\partial q_{k2}} - f\dfrac{\partial(C_{232})}{\partial q_{k2}}\end{bmatrix} \tag{59}$$

$$\beta = (C_{312}(X_i - X_{C2}) + C_{322}(Y_i - Y_{C2}) + C_{332}(Z_i - Z_{C2})) \tag{60}$$

In order to find the partial derivatives of the colinearity equations with respect to the MRP vector, the partial derivative of the direction cosine matrix with respect to the MRP vectors must be found. C_{PQ2} denotes a cell in the cosine matrix where p=1,2,3 and q=1,2,3. Using the orthogonal direction cosine matrix from Chapter II, S_{PQ} denotes the numerator, and T denotes the square root of the denominator.

$$[C] = \frac{1}{T^2}\begin{bmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{bmatrix} \tag{61}$$

$$[S] = \begin{bmatrix} 4(q_1{}^2 - q_2{}^2 - q_3{}^2) + \sigma^2 & 8q_1q_2 - 4q_3\sigma & 8q_1q_3 - 4q_2\sigma \\ 8q_2q_1 - 4q_3\sigma & 4(-q_1{}^2 + q_2{}^2 - q_3{}^2) + \sigma^2 & 8q_2q_3 - 4q_1\sigma \\ 8q_3q_1 - 4q_2\sigma & 8q_3q_2 - 4q_1\sigma & 4(-q_1{}^2 - q_2{}^2 + q_3{}^2) + \sigma^2 \end{bmatrix} \tag{62}$$

$$T = (1 + \underline{q}^T \underline{q}) \tag{63}$$

$$\sigma = (1 - \underline{q}^T \underline{q}) \tag{64}$$

The derivative of each cell in the cosine matrix is found using the chain rule.

$$\frac{\partial(C_{PQ2})}{\partial q_{k2}} = \frac{1}{T_2^2}\left(\frac{\partial(S_{PQ2})}{\partial q_{k2}} - \left(\frac{2T_2 S_{PQ2}}{T_2^2}\right)\left(\frac{\partial(T_2)}{\partial q_{k2}}\right)\right) \tag{65}$$

To obtain the final form of the partial derivative, the derivative of T with respect to q_k is substituted into the chain rule.

$$\frac{\partial(T_2)}{\partial q_{k2}} = 2q_{k2} \tag{66}$$

$$\frac{\partial(C_{PQ2})}{\partial q_{k2}} = \frac{1}{T_2^2}\left(\frac{\partial(S_{PQ2})}{\partial q_{k2}} - 4C_{PQ2}T_2 q_{k2}\right) \tag{67}$$

The derivative of the S matrix with respect to each q_k is defined.

$$\frac{\partial(S_2)}{\partial q_{12}} = \begin{bmatrix} 4T_2 q_{12} & 8q_{22} - 8q_{32}q_{12} & 8q_{32} + 8q_{22}q_{12} \\ 8q_{22} + 8q_{32}q_{12} & 4(T_2 - 4)q_{12} & 4\sigma_2 - 8q_{12}^2 \\ 8q_{32} - 8q_{22}q_{12} & -4\sigma_2 + 8q_{12}^2 & 4(T_2 - 4)q_{12} \end{bmatrix} \tag{68}$$

$$\frac{\partial(S_2)}{\partial q_{22}} = \begin{bmatrix} 4(T_2 - 4)q_{22} & 8q_{12} - 8q_{32}q_{22} & -4\sigma_2 + 8q_{22}^2 \\ 8q_{12} + 8q_{32}q_{22} & 4T_2 q_{22} & 8q_{32} - 8q_{22}q_{12} \\ 4\sigma_2 - 8q_{22}^2 & 8q_{32} + 8q_{22}q_{12} & 4(T_2 - 4)q_{22} \end{bmatrix} \tag{69}$$

$$\frac{\partial(S_2)}{\partial q_{22}} = \begin{bmatrix} 4(T_2 - 4)q_{32} & 4\sigma_2 - 8q_{32}^2 & 8q_{12} + 8q_{32}q_{22} \\ -4\sigma_2 + 8q_{32}^2 & 4(T_2 - 4)q_{32} & 8q_{22} - 8q_{32}q_{12} \\ 8q_{12} - 8q_{32}q_{22} & 8q_{22} + 8q_{32}q_{12} & 4T_2 q_{32} \end{bmatrix} \tag{70}$$

Measured Camera Position and Attitude

In the third strategy, sensors are used to reduce uncertainty and give measured locations and attitudes of the cameras with respect to the origin. A possible sensor that can be employed is a modulating retro-reflector, which is described in more detail in Appendix B. The location and attitude of the second camera is still considered unknown, but the measured values introduce new equations into $\Delta \underline{y}_C$:

$$\Delta \underline{y}_C = \begin{bmatrix} \Delta x_{ij} \\ \Delta y_{ij} \\ \Delta X_{C2} \\ \Delta Y_{C2} \\ \Delta Z_{C2} \\ \Delta q_{12} \\ \Delta q_{22} \\ \Delta q_{32} \end{bmatrix} = \begin{bmatrix} \tilde{x}_{ij} - F(X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, q_{j1}, q_{j2}, q_{j3}, f_j) \\ \tilde{y}_{ij} - G(X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, q_{j1}, q_{j2}, q_{j3}, f_j) \\ \tilde{X}_{C2} - \hat{X}_{C2} \\ \tilde{Y}_{C2} - \hat{Y}_{C2} \\ \tilde{Z}_{C2} - \hat{Z}_{C2} \\ \tilde{q}_{12} - \hat{q}_{12} \\ \tilde{q}_{22} - \hat{q}_{22} \\ \tilde{q}_{32} - \hat{q}_{32} \end{bmatrix} \tag{71}$$

$$known: X_{C1}, Y_{C1}, Z_{C1}, q_{11}, q_{21}, q_{31}, f_j$$
$$unknown: X_i, Y_i, Z_i, X_{C2}, Y_{C2}, Z_{C2}, q_{12}, q_{22}, q_{32}$$

Using these additional sensors increases the number of rows in the *H*-matrix by 6, so the dimension for the third application is 4*p+6 by 3*p+6. As in the first strategy, there is no minimum number of points required to ensure the system is not indeterminate, because there are always more equations than unknown values. Figure 12 contains the new *H*-matrix, with the same color-coding as in Figure 10 and 11.

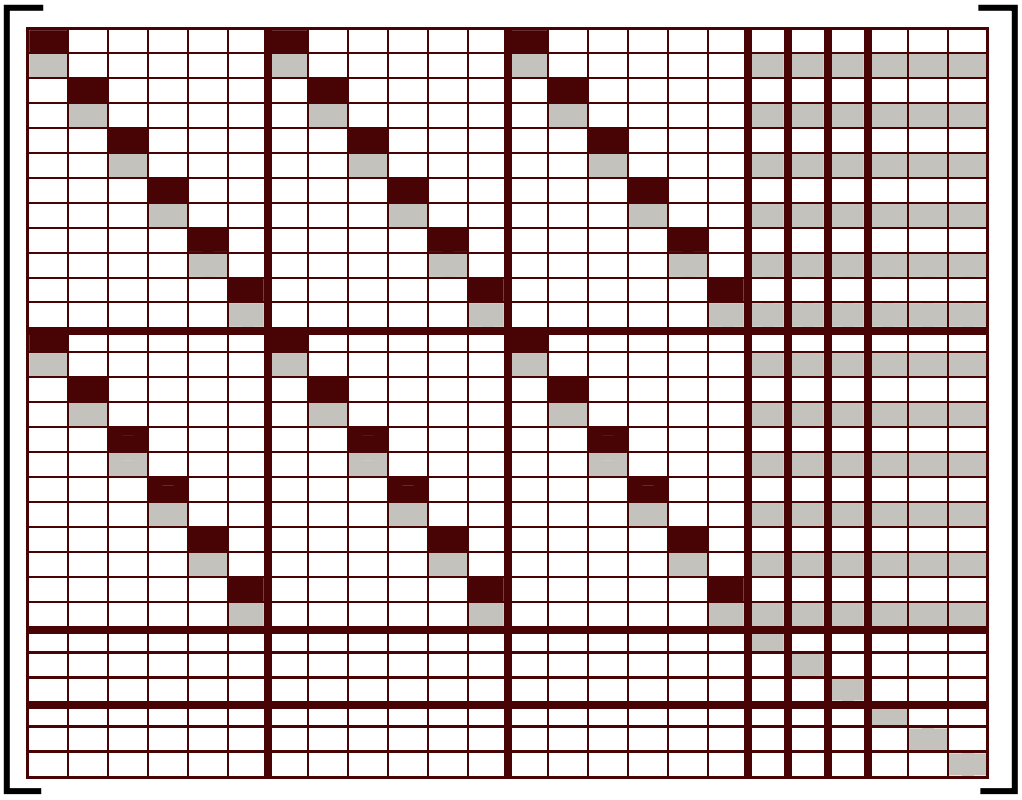


Fig. 12 Populated *H*-Matrix for Strategy 3.

As shown in the figure, the gradient of the measured values with respect to itself is 1 and with respect to all other values (X_i, Y_i, Z_i, X_{cj}, Y_{cj}, Z_{cj}, and q_{kj}) is 0.

$$\frac{\partial(\tilde{X}_{C2})}{\partial X_{C2}} = 1 \tag{72}$$

$$\frac{\partial(\tilde{Y}_{C2})}{\partial Y_{C2}} = 1 \tag{73}$$

$$\frac{\partial(\tilde{Z}_{C2})}{\partial Z_{C2}} = 1 \tag{74}$$

$$\frac{\partial(\tilde{q}_{12})}{\partial q_{12}} = 1 \tag{75}$$

$$\frac{\partial(\tilde{q}_{22})}{\partial q_{22}} = 1 \tag{76}$$

$$\frac{\partial(\tilde{q}_{32})}{\partial q_{32}} = 1 \tag{77}$$

Results

The algorithms in the previous chapters are applied to an extremely thin surface that is 5m long (in the x-direction), 7m deep (in the y-direction), and located 5m from the camera lens (in the z-direction). Each camera has a focal length of 10mm, and there is a distance of 5m between the two cameras. There is a 29x29 matrix of points projected onto the surface of the antenna. A measurement error of $5*10^{-3}$m is assumed for the sensor in the third strategy.

The actual surface (Figure 13) appears identical to the first strategy (Figure 14) and third strategy (Figure 16) surface. The second strategy (Figure 15) appears very similar to the actual surface, except Figure 15 shows a much higher peak at the back right hand corner than in the actual surface. The estimated cameras' locations are represented on the figures as triangles.

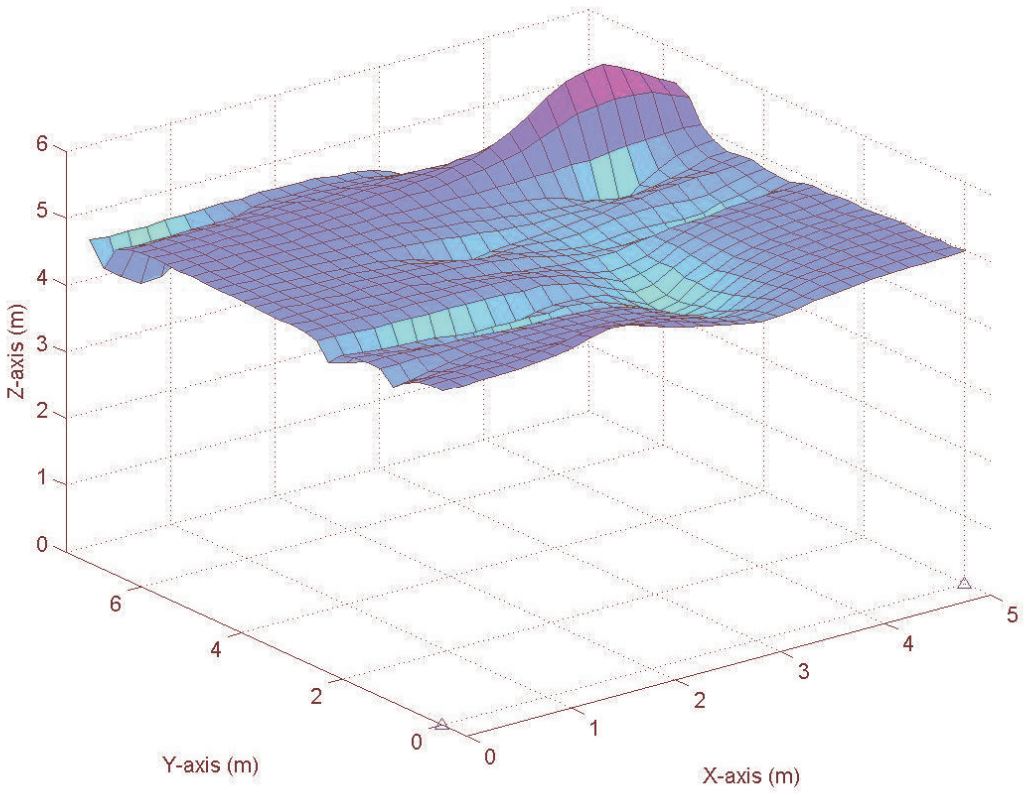


Fig. 13 Actual Surface.

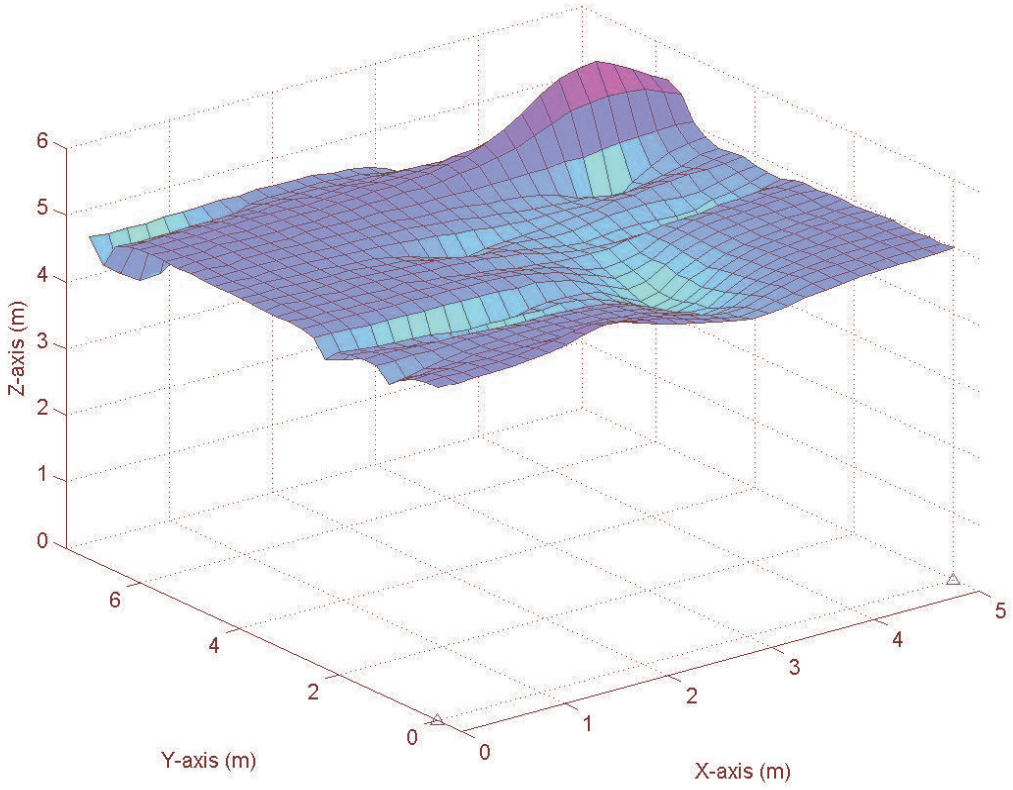


Fig. 14 Absolute Camera Knowledge.

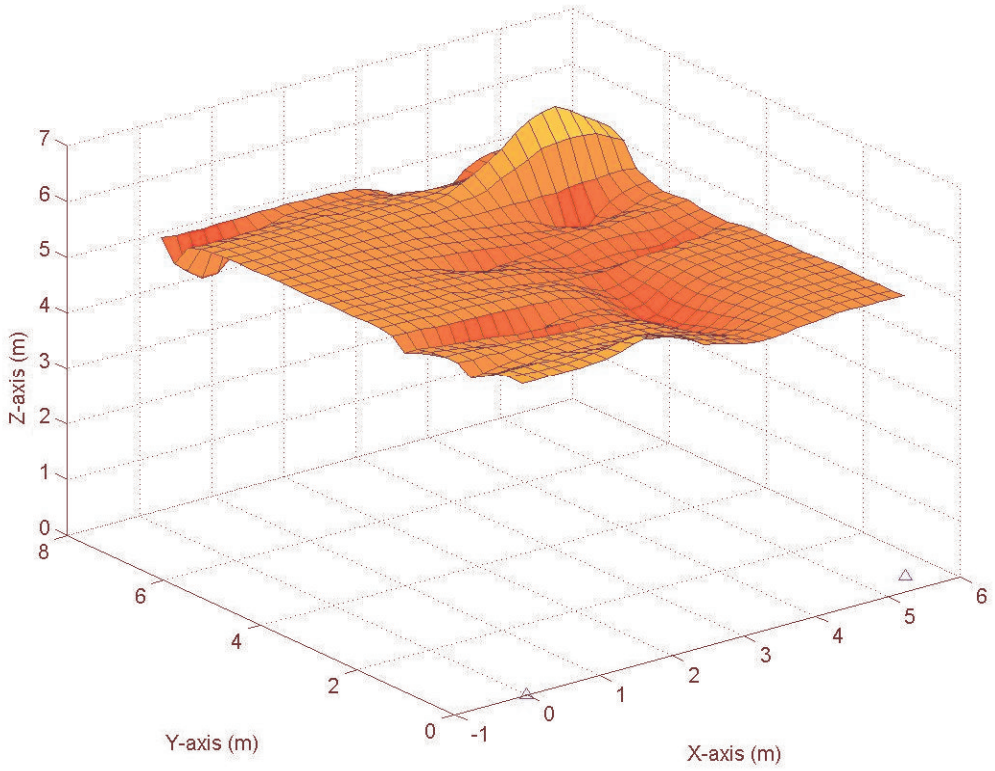


Fig. 15 No Camera Knowledge.

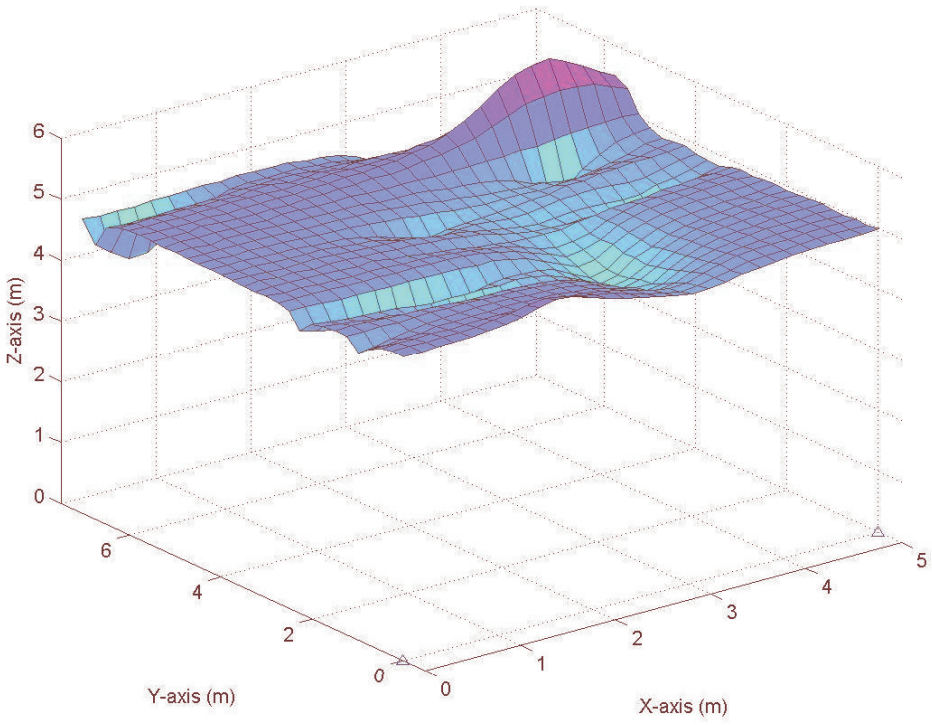


Fig. 16 Measured Camera Knowledge.

Notice on Figure 15, camera 2 is estimated to be located at 5.4m, an error of 0.4m. In the third strategy, camera 2 is as estimated to be at 5.005m, an error of only 5mm. The maximum error of the norm of each point's location associated with each strategy (Figure 17) is significantly small, with a maximum of 0.007m for strategy 1 (Figure 18), 0.65m for strategy 2 (Figure 19), and 0.007m for strategy 3 (Figure 20). In Figure 13, the point number is assigned from the front to the back of the panel along the y-axis (y=0 to y=7m) for each row of points on the x-axis (from x=0 to x=5m). This collection method explains the saw-tooth pattern seen in strategy 2. The largest errors are located at the back of the panel, which implies that the further the point is from the camera, the higher the estimation error. The random distribution of error in strategy 1 and 3 illustrates that the absolute or measured knowledge of the cameras' locations and

attitudes do not cause a change in the accuracy of the points based on their distance to the camera, as is the relationship in strategy 2.

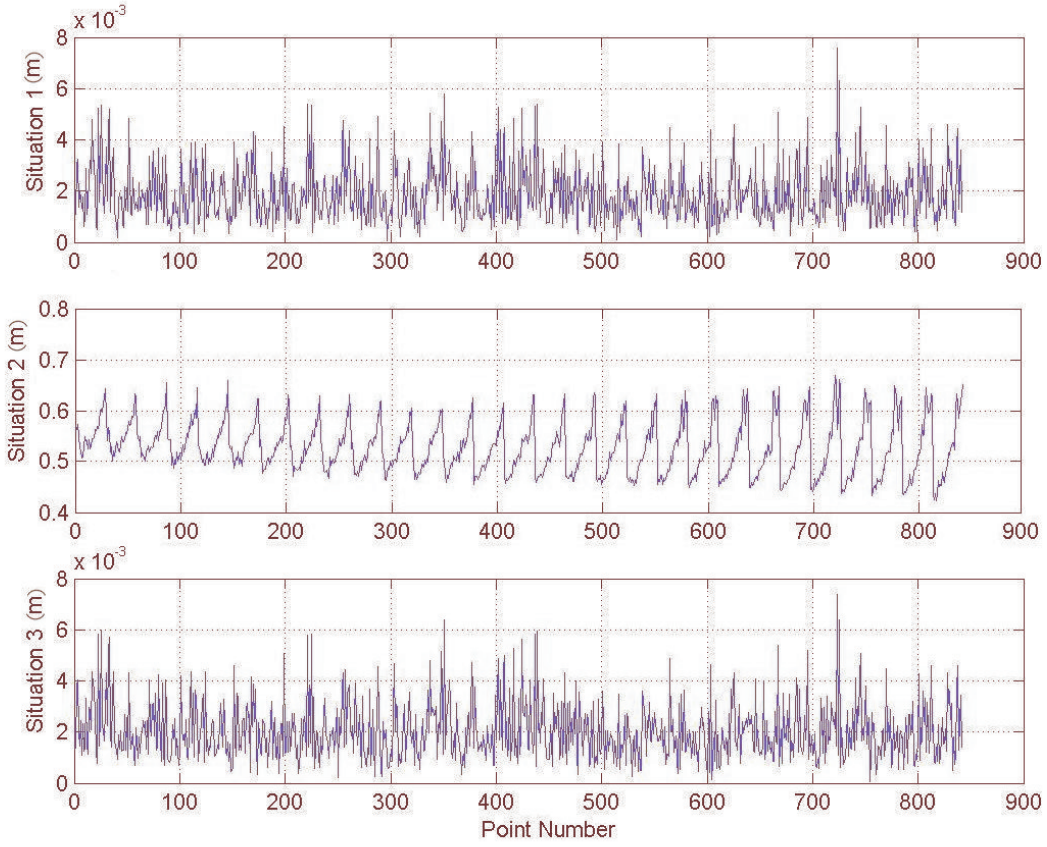


Fig. 17 Estimation Point Location Error.

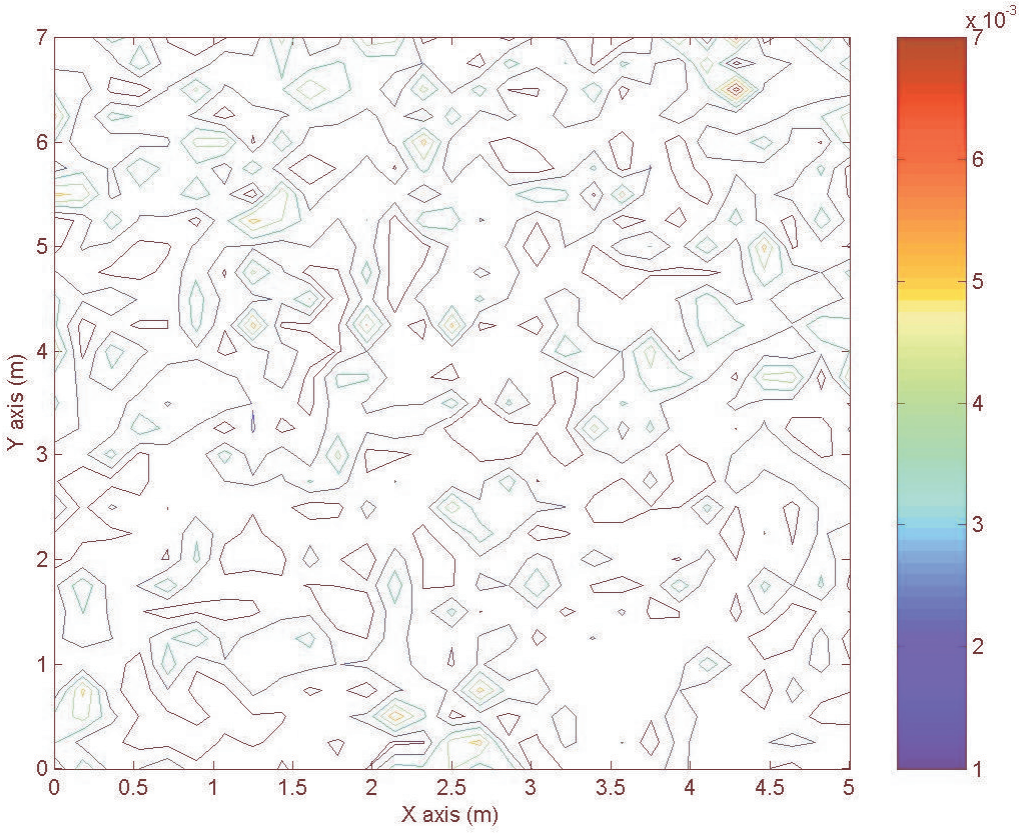


Fig. 18 Contour Plot of Estimation Error for Strategy 1.

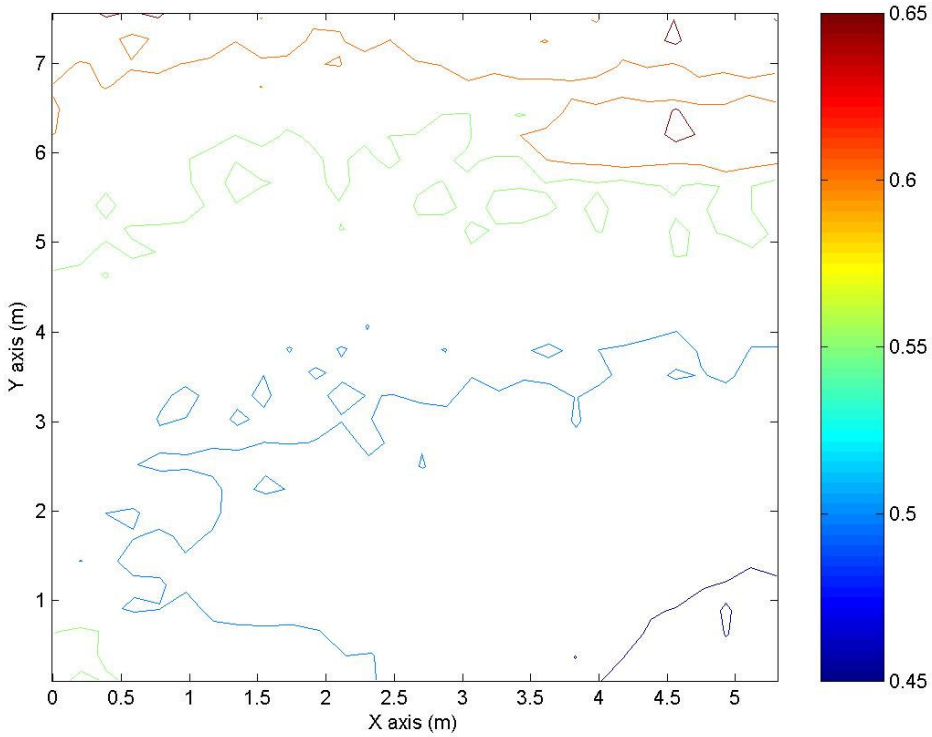


Fig. 19 Contour Plot of Estimation Error for Strategy 2.

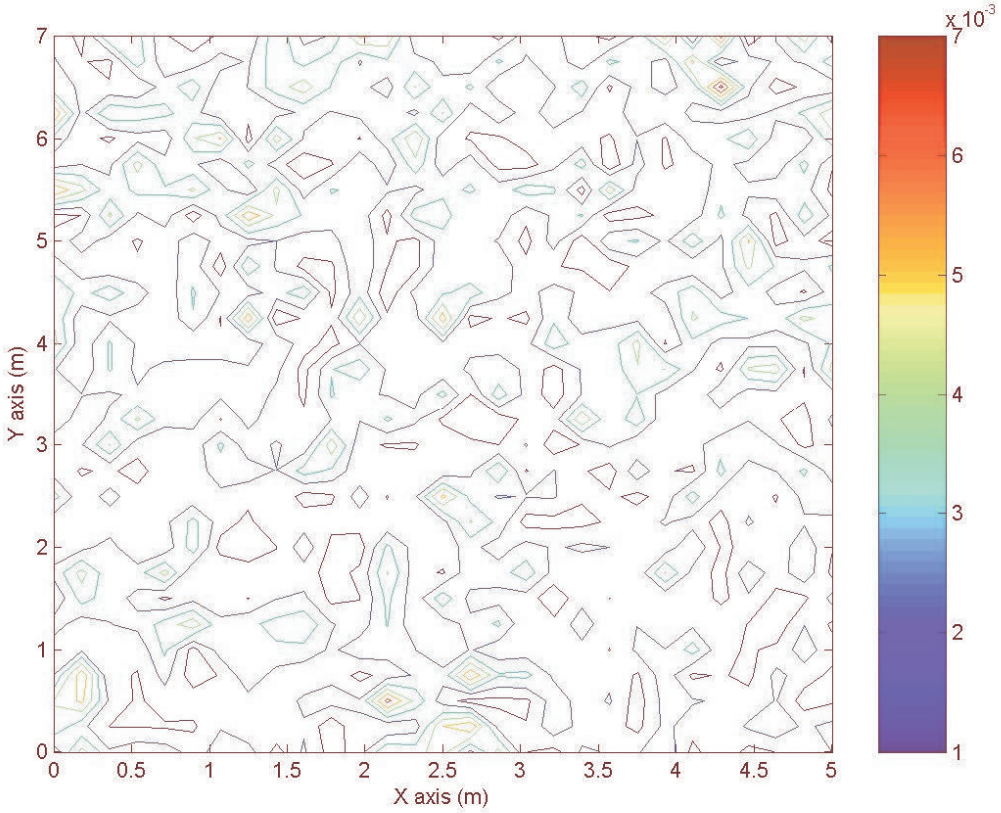


Fig. 20 Contour Plot of Estimation Error for Strategy 3.

Considering the only information known in the second strategy is that the first camera is located at the origin, a maximum norm of the point location error of 0.65m is acceptable. From the figures, the methods employed by this thesis clearly create successful models because the largest errors for strategies 1 and 3 are measured in millimeters.

CHAPTER IV

GLOBAL ENVIRONMENT

Description of Surface

The entire SBR antennae surface that is modeled in the global environment is 100m long (x-direction) and 7m wide (y-direction). The cameras and laser dot projectors are alternately attached to a rigid beam, spanning the length of the antenna. The flexible antenna is attached to the beam by a metal frame, separating the cameras and the antenna by approximately 5m. Each camera has a 50mm x 50mm focal plane, a 25mm focal length, and a 90° field of view. To minimize the number of cameras, each camera is part of two adjacent stereo-triangulation pairs. The maximum angle between a camera optical axis and the furthest offset point to be imaged is assumed to be 45°. Based on these specifications for the cameras, every pair of cameras is separated by a fixed distance of 5m; twenty one cameras are required to image the entire antenna. A laser dot projector located exactly half-way between each pair of cameras produces a matrix of points onto the antenna. A portion of the global environment is shown in Figure 21.

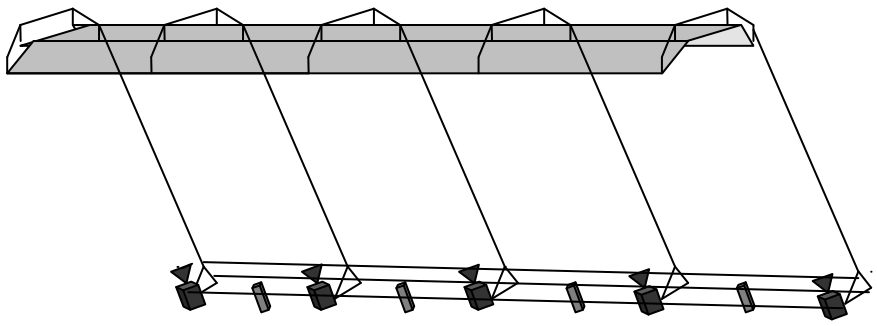


Fig. 21 Global Environment.

Combining Local Environments

The entire antenna surface is estimated by using the equations developed from the Local Environment, for each stereo-pair. The equations could be adapted to include many cameras which would put the entire surface into one nonlinear least squares program. While this would be possible for the first strategy, it increases the computational complexity because the inverse of the *H*-matrix is required. The Gaussian nonlinear least squares algorithm could not be adapted for the entire surface to be approximated at once for the second and third strategies because each camera's location and attitude is found using the location of the previous camera in the pair. There would be no previous camera information for adapted strategies because in this hypothetical algorithm, there are not any pairs, only a large number of cameras.

For computational simplicity and feasibility, the algorithm iterates each pair of cameras. With many cameras imaging the same surface, the overlapping surfaces are referred to as panels (Figure 22).

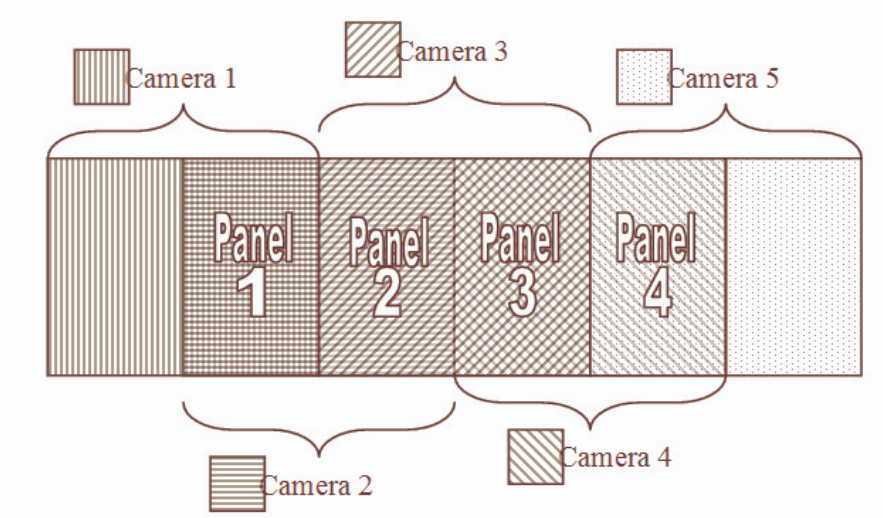


Fig. 22 Camera Overlap.

Iterating each local pair produces a daisy-chain effect which propagates errors in the camera location and attitude as well as causes the furthest cameras from the origin to have the highest errors. For this reason, the origin is set in the middle of the beam and not at one of the ends of the beam.

Results

The following application uses half of the beam, starting from camera 1 at the origin, with its axes aligned with the principle axes, to camera 11 at 50m. Using 64 points per panel, there are a total of 640 points to estimate the location. A Gaussian noise distribution with a standard deviation of 10^{-6}m is added to all measurements which includes the surface points' location in image space as well as camera locations and attitudes.

The actual surface (Figure 23) is compared to the estimated surface which assumes absolute camera position and attitude are known (Figure 24). It is extremely accurate and appears identical. The actual surface compared to the estimated surface in strategy 2 where there is no knowledge of the camera information (Figure 25) appears similar when observing points close to the origin, and then begins to diverge dramatically as the distance from the points to the origin increases. As with the first strategy, the third strategy where the camera information is measured (Figure 26) appears identical to the original surface.

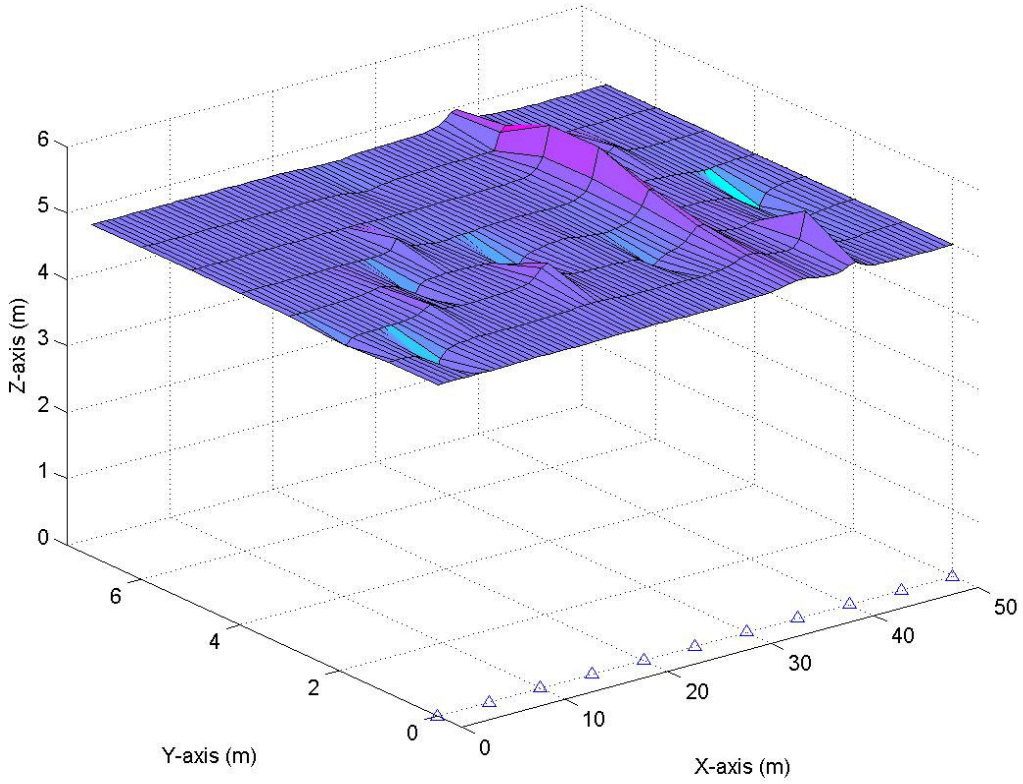


Fig. 23 Original Surface.

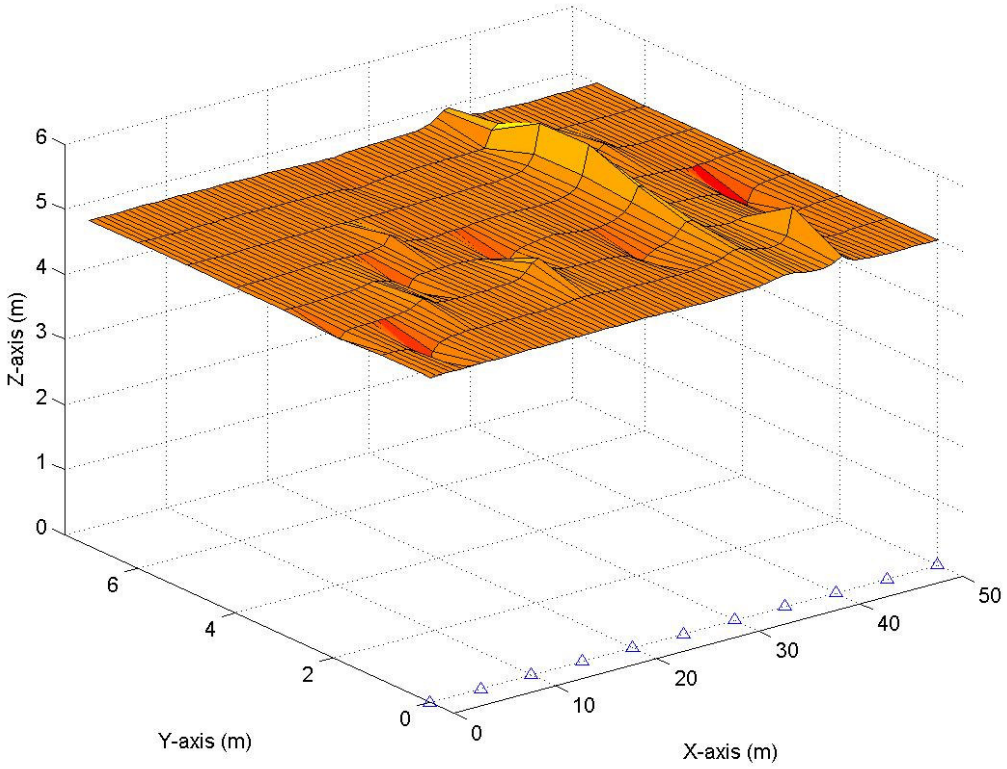


Fig. 24 Strategy 1 Estimated Surface.

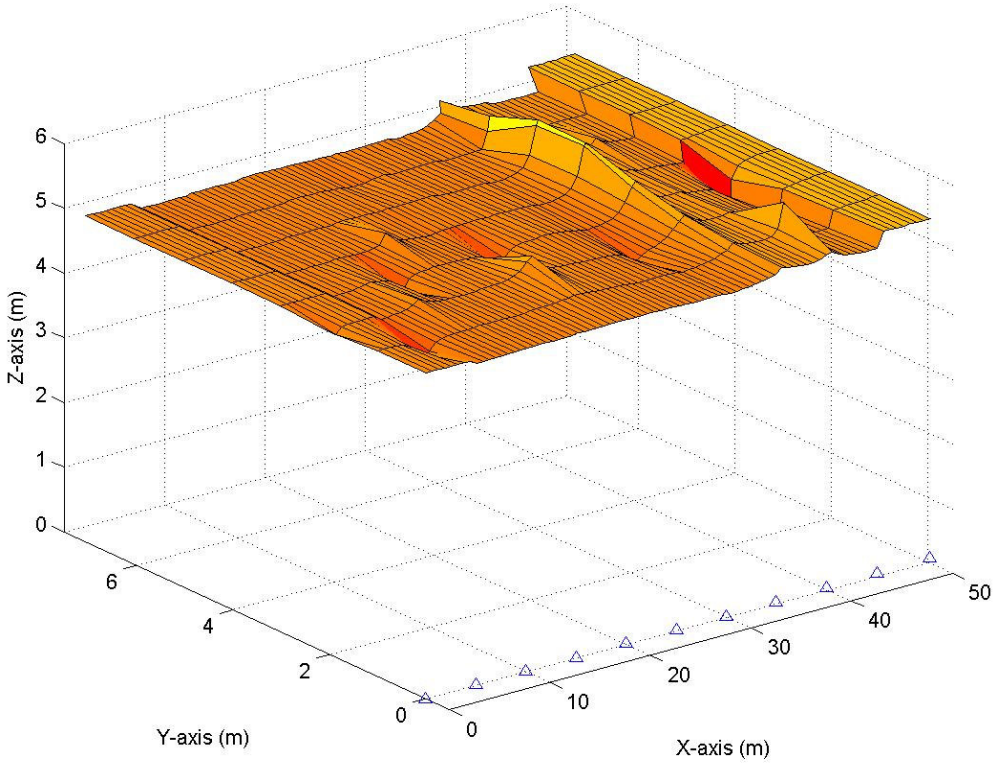


Fig. 25 Strategy 2 Estimated Surface.

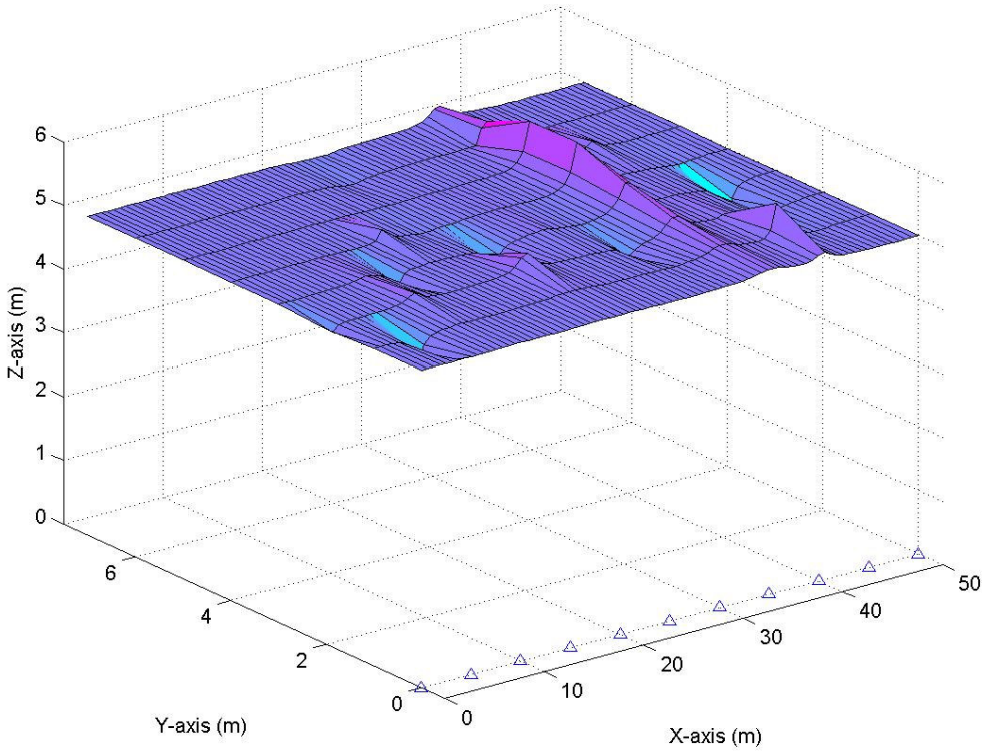


Fig. 26 Strategy 3 Estimated Surface.

Figure 27 displays the norm of the point location errors for each strategy. Every line represents a panel of sixty four points for that strategy. The estimation error for the first strategy is very small, with a maximum error of 0.007 m. The second strategy has a maximum point error of 0.24m, almost thirty two times greater than the first strategy. In the third strategy, the maximum error reaches 0.01m; this is only about 1.5 times greater error than the first strategy. Also, the error for the first and third strategy occurs randomly. The second strategy has an increasing error with each panel. The highest error for each panel occurs at the back of the panel and for the entire surface occurs at the panel furthest from the origin. This pattern confirms the daisy-chain effect that is present in the second strategy.

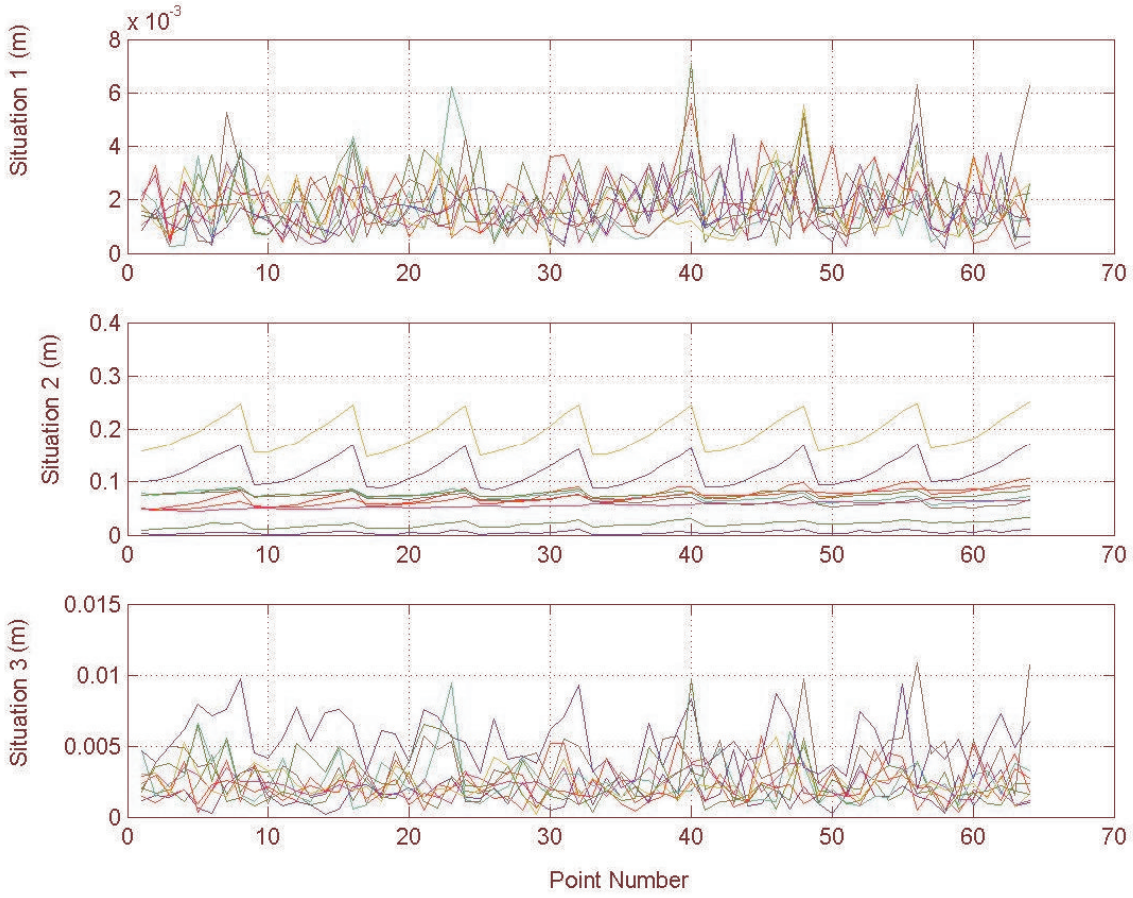


Fig. 27 Point Location Errors.

A contour plot of the point errors in strategy 1 (Figure 28) shows that the errors are highest at the back of the surface, the point furthest from the camera along the y-axis, but evenly distributed along the x-axis. A contour plot of the point errors in the second strategy (Figure 29) displays a more dramatic error distribution. The first few panels contain an error of 0.06m, and this error increases to 0.1m for most of the panel. There is a large spike in error at the back of the panel where there is a peak value in the actual surface. The contour plot of point errors for strategy 3 (Figure 30) reveals that the highest error occurs in the rear corner of the surface which shows the errors compound with each additional camera.

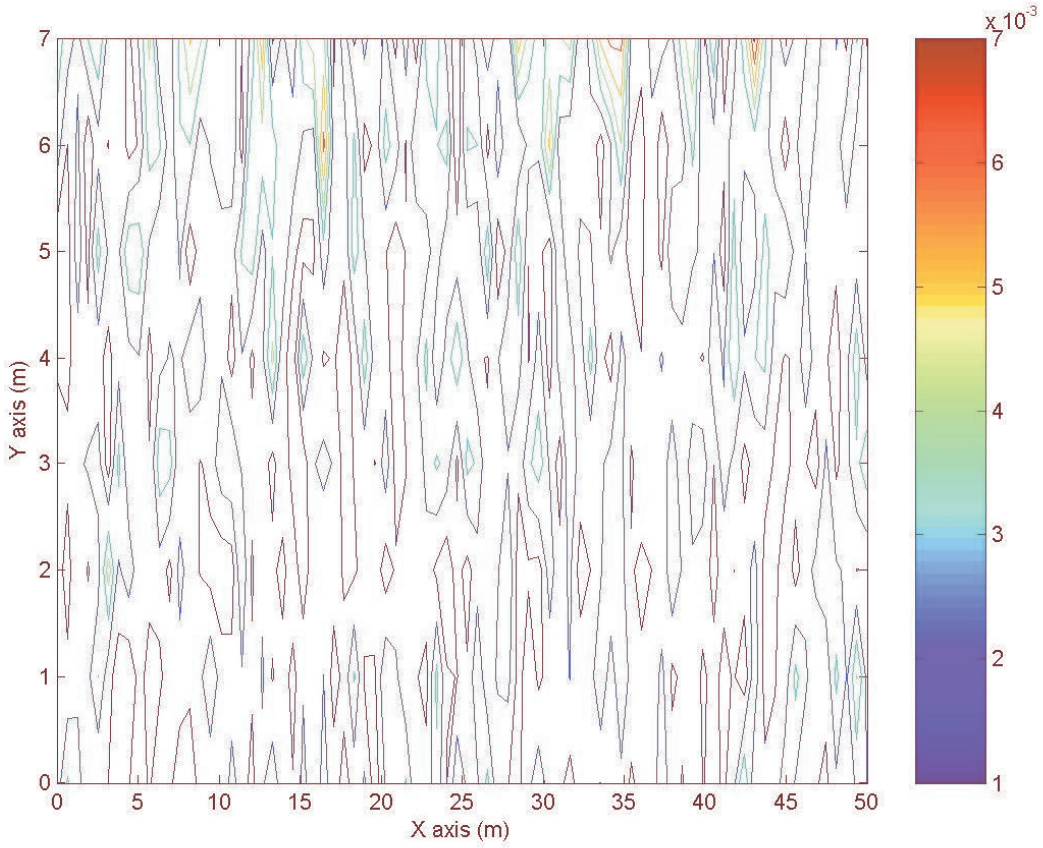


Fig. 28 Strategy 1 Point Errors Contour Plot.

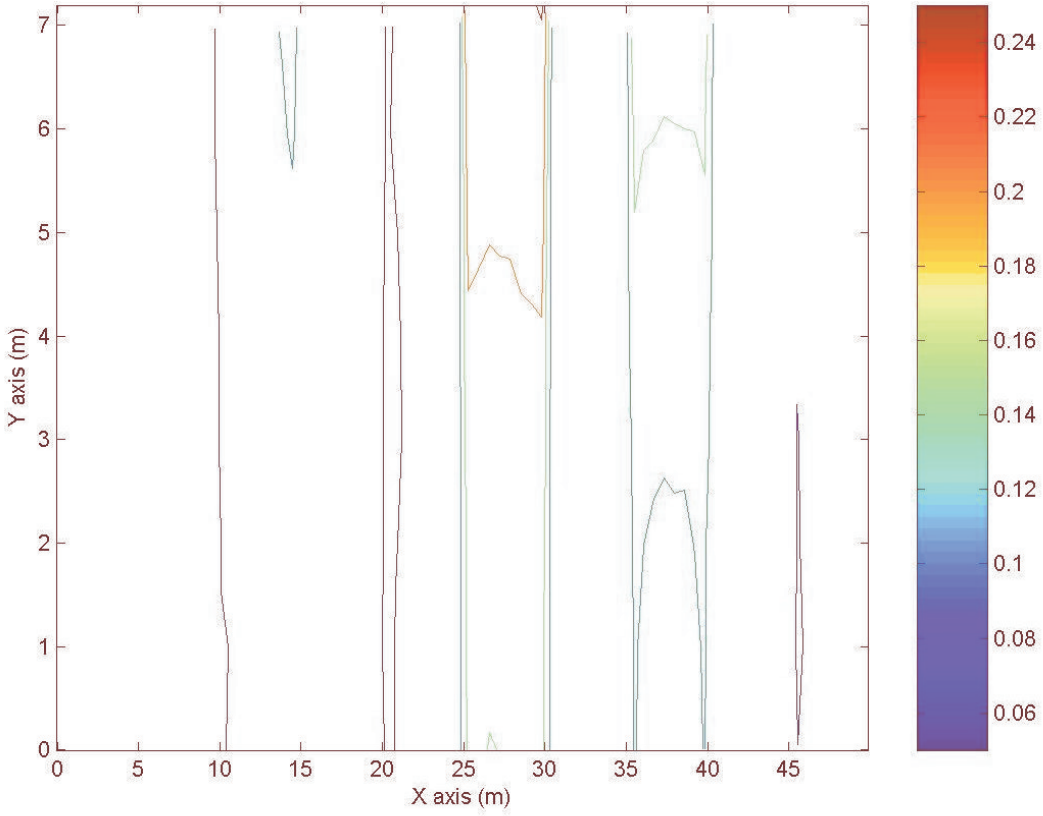


Fig. 29 Strategy 2 Point Errors Contour Plot.

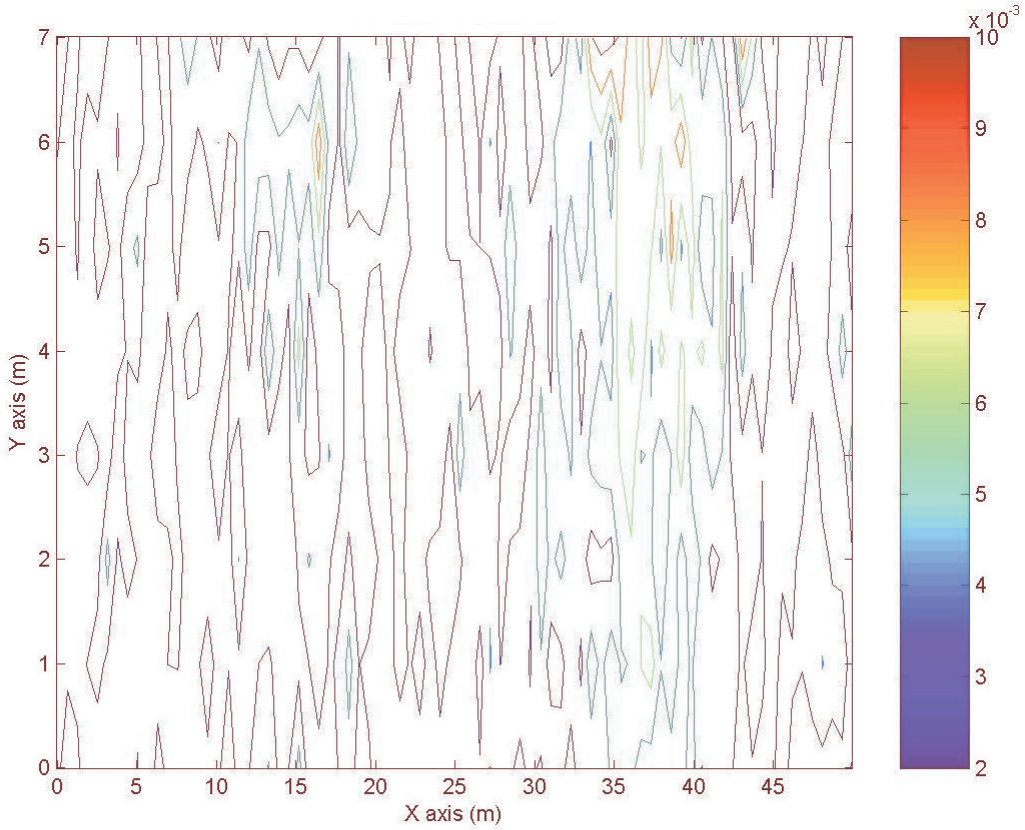


Fig. 30 Strategy 3 Point Errors Contour Plot.

The norm of the camera location errors for the second and third strategies are displayed in Figure 31. As predicted, the errors in the second strategy compound with each camera, which causes a maximum error of 0.26m at the last cameras. In strategy 3, the error is much smaller and more random; there are three peak values distributed along the x axis. The maximum error is 2.6mm, 100 times smaller than in the second strategy. This error is attributed to the noise in the measurements. A priori values for the cameras' locations and orientations in the estimation process produce a much more accurate result.

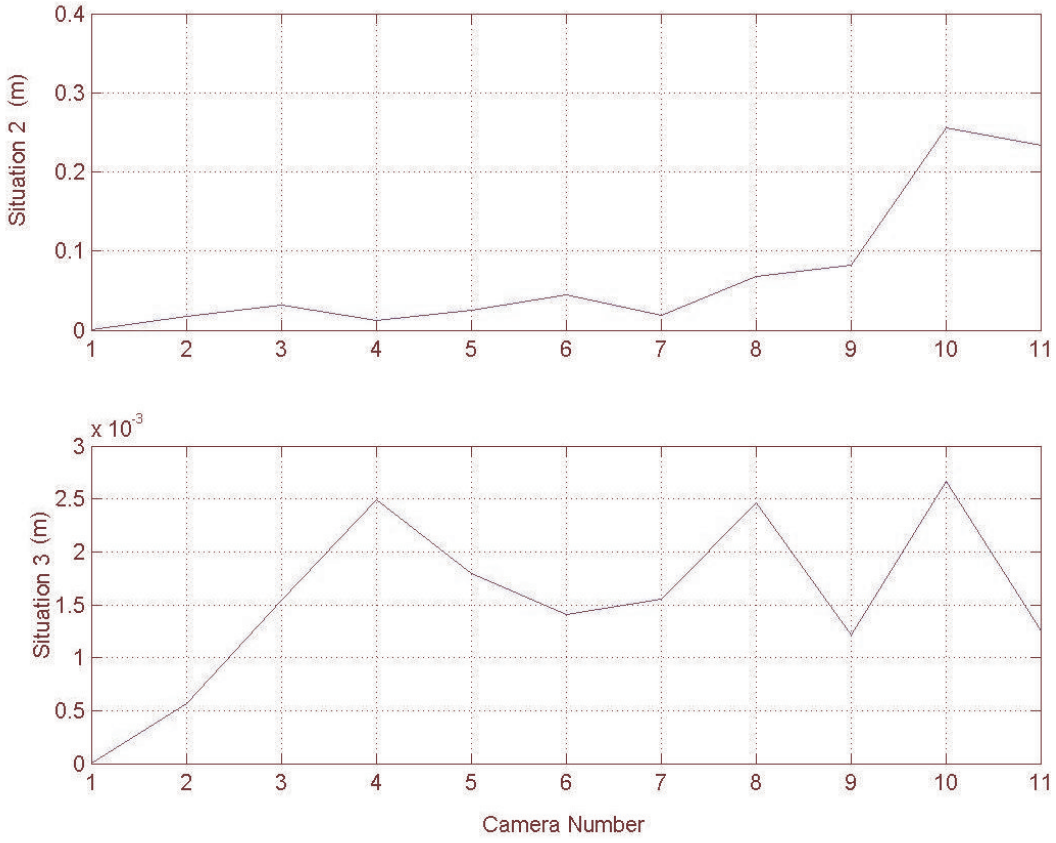


Fig. 31 Camera Location Errors.

The norm of the orientation errors of the cameras are displayed in Figure 32 for the second and third strategies. The attitude error in the second strategy increases as the camera number increases. The maximum error 0.0024rad occurs at camera 9. The third strategy has a much more random result and contains a much smaller maximum error of $3.7*10^{-7}$rad that occurs at the second camera. Again, a priori knowledge from a sensor decreases the error for a more accurate estimate of camera orientation.

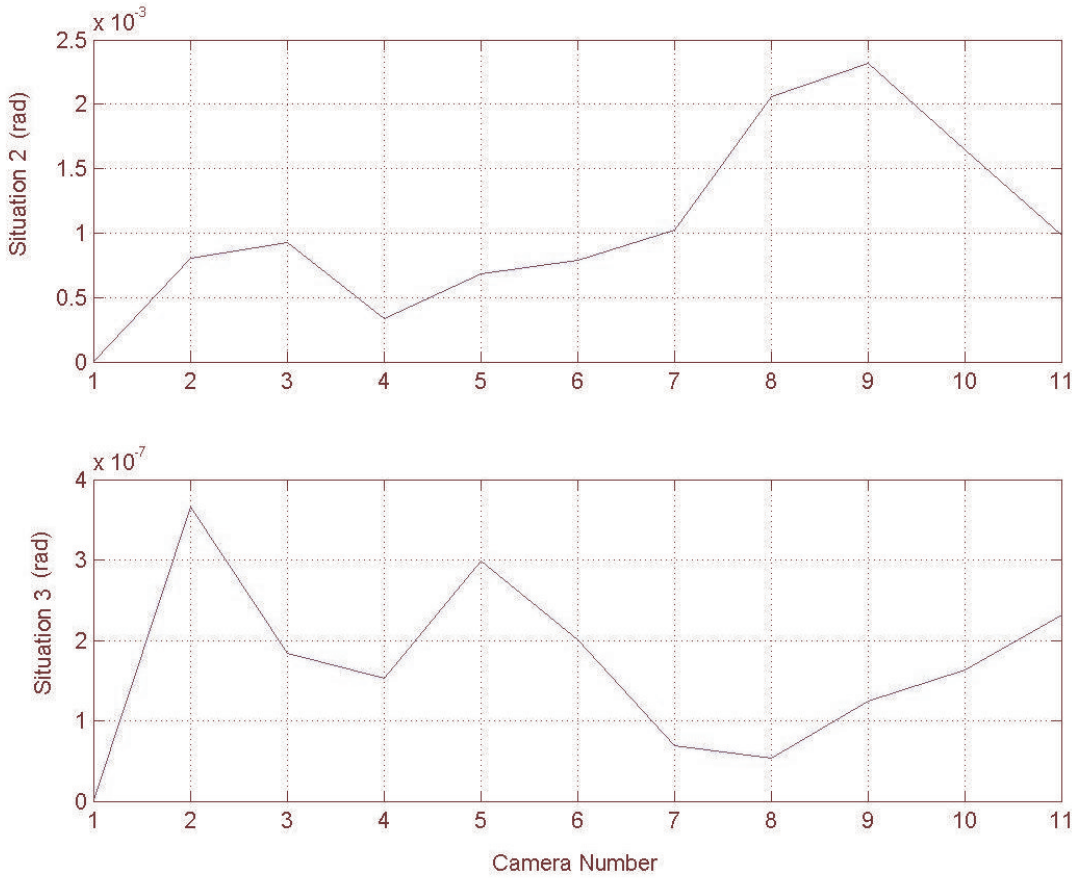


Fig. 32 Camera Orientation Errors.

It is a clear assumption that the more points per panel, the more accurately the surface is approximated. However, there is no correlation between the number of total points and the maximum estimation error (Figure 33) for point error, camera location, or camera orientation error. Figure 33 displays the maximum error from 10 surfaces for each number of points.

The line representing the maximum point location errors for the first and third strategies appear colinear which displays the similarity of the results. Also, the accuracy of the first and third strategies appears unaffected by the number of total points. For the second strategy, there is no correlation between the number of points and the accuracy as

it appears to fluctuate with each strategy. The error is therefore not affected by the number of points but by the surface shape.

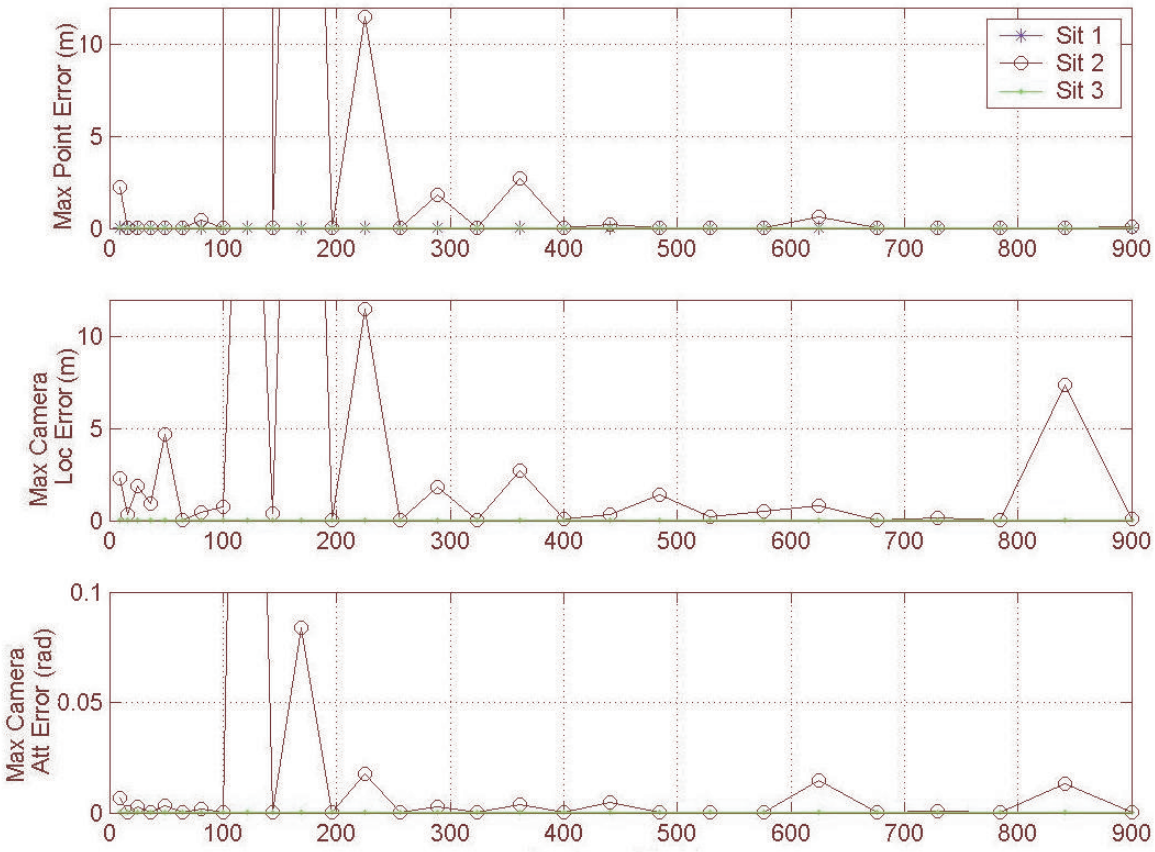


Fig. 33 Variation of the Number of Points.

When the noise value for the measured values is altered (Figure 34), it is expected that the errors increase slightly in strategy 1, greatly in strategy 2 because the camera location and orientation are very sensitive, and noticeably in strategy 3 because in addition to the location of the points in the image plane, the camera location and orientation are measured values.

The errors in Figure 34 are the maximum error from 10 surfaces for each change in the standard deviation of the noise value. The errors for strategy 1 increase linearly as expected. The errors in strategy 2 increase dramatically after a noise value of 10^{-4}. The errors in strategy 3 have almost the exact values as strategy 1 until a measurement error of 10^{-3}. For each of the errors, the second strategy has a larger error than the first or third strategy. Clearly, each strategy is affected by the noise value, so accuracy drives the type of sensors used for measurement.

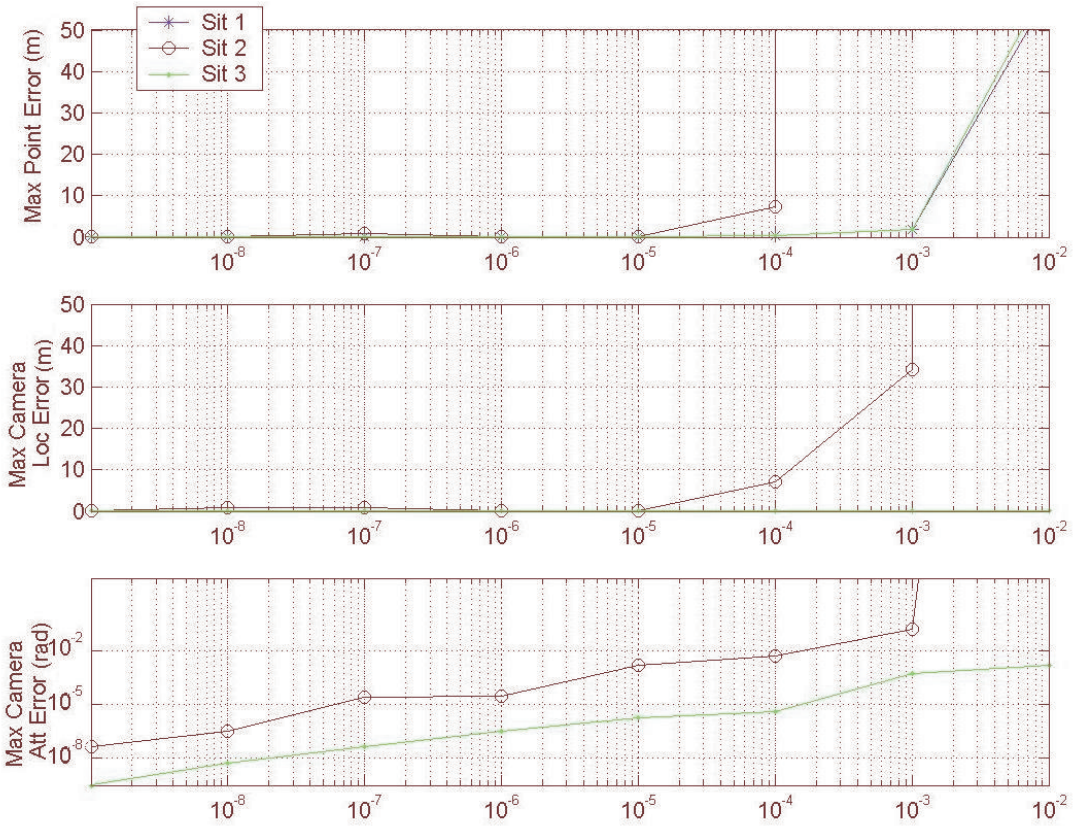


Fig. 34 Variation of the Noise Value.

As expected, strategy 1 produces the most accurate results, but for an application where the cameras' locations and attitudes are not exactly known (measurement error exists), the surface estimate is more unrealistic. Strategy 2 produces a less accurate

estimate, but it is the easiest to implement and can be realistically implemented for any application. Strategy 3 requires additional sensors, but it produces an extremely accurate and realistic answer. Appendix A contains a linearization method of the colinearity equations to produce a linear least squares estimation for strategy 1.

CHAPTER V

SUMMARY AND CONCLUSION

Every strategy produces a different result when the number of points, number of cameras, noise value, or surface configuration is changed. The application of any of the strategies is dependant on these variables, which implies that the strategies are adaptable to many different applications based on the constraints. For example, if the number of points that the laser dot projector produces increases, the computational power required also increases, but the result produces a much smoother, more realistic curve. To increase the number of cameras adds redundancy, but more computation is required, and the volume and weight consumed by the system increases. The accuracy is also affected by these variables. Accuracy decreases for strategies 2 and 3 as the number of cameras increase. The larger noise value results in a decreased accuracy. A surface with very steep peaks causes the weakest contour geometry. The number of points does not seem to affect the accuracy though more exhaustive research on the subject would be valuable.

For the Local Environment, strategy 1 produces the most accurate results, followed by strategy 3, and finally strategy 2. When the Local Environment is iterated for each panel as in the Global Environment, strategy 1 is unaffected. Strategy 3 is mildly affected, but the magnitude of the error in strategy 2 increases dramatically. These increases are due to the compounding of errors proportional to the distance that a point or camera is from the origin. Also, when the camera location and attitude start to incur errors, the estimation accuracy of the points' locations begins to drop. A small error in the attitude can have a large influence on the accuracy of the points' locations.

As expected for the Global Environment, the highest accuracy is found in strategy 1 where the least number of unknowns exist. There are very few applications for this type of a system however. For a more realistic application where the cameras' locations and attitudes are not exactly known (measurement error exists), as in strategy 3, the surface estimate is almost as accurate as strategy 1. The downfall to strategy 3 is that more equipment is required, which is a limiting factor for space applications as it

increases the cost, weight, and volume of the system. The most easily applied and realistic solution is strategy 2, but the cost of an easier to implement strategy is the estimation accuracy. Considering the only known value for strategy 2 is the location and attitude of camera 2, the accuracy obtained is significant.

A great advance that could be realized if more time is spent researching linearization would be to find a closed form solution for the colinearity equations. While it is advantageous to have strategy 1 in a linear least squares algorithm (Appendix A), as opposed to a nonlinear least squares algorithm, it would be nice to adapt the linearization for strategy 2 and 3. Also, it would be interesting to analyze the covariance analysis of each point which is the value of how accurate the algorithm believes it is at each point. This would entail comparing the overall accuracy to the covariance matrix ($P=H^TWH$).

Chapter IV explains that iterating each pair of cameras is required for strategies 2 and 3 because the position and orientation of one of the cameras in each pair is required. A further development for this algorithm would be to create a bulk estimation of the entire surface after each panel is estimated. This could help correct the daisy-chain effect present in strategies 2 and 3, although it would increase computational complexity.

All of the strategies developed in this thesis provide a lightweight, low-volume response to the need for new sensors in space. This program can be easily adapted and optimized for a number of applications.

REFERENCES

[1]Valkenburg, R.J., and McIvor, A.M. "Accurate 3D Measurement Using a Structured Light System," Report 576, Industrial Research Limited, June 1996.

[2]Morris, D. D., and Kanade, T. "Image-Consistent Surface Triangulation," *2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'00),* Vol 1, No. 4, 2000, p. 1332.

[3]Junkins, J.L., Hughes, D., Wazni, K., and Pariyapong, V., "Vision-Based Navigation for Rendezvous, Docking, and Proximity Operations," *Advances in Astronautical Sciences,* Vol 101, 1999, pp 203-220

[4]Schaub, H.P., and Junkins, J. L., *Analytical Mechanics of Space Systems,* AIAA Education Series, AIAA, Reston, VA, 2003, pp 107-110.

[5]Junkins, J.L., *An Introduction to Optimal Estimation of Dynamical Systems,* Sijthoff & Noordhoff International publishers, Netherlands, 1978, pp 91-104.

[6]Crassidis, J. L., and Junkins, J. L., *Optimal Estimation of Dynamic Systems,* CRC Press LLC, Boca Raton, FL, 2003, pp 7 – 28.

[7]Mortari, D., Rojas, M., and Junkins, J.L., "Attitude and Position Estimation from Vector Observations," 14th AAS/AIAA Space Flight Mechanics Meeting, Advances in Astronautical Sciences 04-140, February 8-12, 2004.

[8]Palm, W. *Introduction to Matlab for Engineers,* McGraw-Hill College, Boston, MA, 1997, pp. 15-18.

[9]Leon, St. *Linear Algebra with Applications,* 6th ed., Pearson Prentice Hall, Upper Saddle River, NJ, 2002, pp. 178-180.

[10]Gilbreath, G.C., Rabinovich, W.S., Meehan, T.J., Vilcheck, M.J., Mahon, R., Ferraro, M., Sokolsky, I., Vasquez, J.A., Bovais, C.S., Cochrell, K., Goins, K.C., Bargehenn, R., Katzer, D.S., Ikossi-Anastasiou, K., Montes, M.J., "Large-Aperture Multiple Quantum Well Modulation Retroreflector for Free-Space Optical Data Transfer on Unmanned Aerial Vehicles," Society of Photo-Optical Instrumentation Engineers, Paper 200371, 2001.

[11]Rabinovich, W.S., Gilbreath, G.C., Bovais, C., Cochrell, K., Burris, H.R., Ferraro, M., Vilcheck, M., Mahon, R., Goins, K., Sokolsky, I., Vasquez, J., Meehan, T., Barbehenn, R., Katzer, D.S., and Ikossi- Ansatasiou, K., ‘‘Infrared data link using a multiple quantum well modulating retro-reflector on a small rotary-wing unmanned airborne vehicle,’’ *Proceedings of the IEEE Aerospace Conference,* Paper 5.054, Mar. 2000.

APPENDIX A

LINEARIZATION OF THE COLINEARITY EQUATIONS

Linearization Method

One method used to linearize a nonlinear equation is to put the equation into a form where a matrix of known values is multiplied by a matrix of unknown values.[7] The null space vector of the matrix of unknown values is then found. Multiplying this vector by a scaling factor produces the unknown vector quantities.

To prevent introducing more nonlinear equations, the attitude parameterization is undefined, so the rotation matrix is in the form C_{rsj}. The following shows the colinearity equations as a function of the rotation matrix for a total number of points (p) and total number of cameras (c):

$$\begin{aligned} x_{ij} &= F(X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, C_{rsj}, f_j) \\ y_{ij} &= G(X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, C_{rsj}, f_j) \end{aligned} \tag{78}$$

$$\begin{aligned} 1 \le i \le p \\ 1 \le j \le c \\ 1 \le r \le 3 \\ 1 \le s \le 3 \end{aligned}$$

A scaling factor (α_{ij}) is required because the null space vector produced could be one of an infinite number of possible vectors that satisfy that matrix. The scaling factor is converted from the vector notation as it was originally described in Eq. (6) to index notation, so it can be applied to the whole surface, not just a single point taken by a single camera as Eq. (6) was originally intended:

$$\alpha_{ij} = \sqrt{\frac{(X_{ij} - X_{Cj})^2 + (Y_{ij} - Y_{Cj})^2 + (Z_{ij} - Z_{Cj})^2}{(x_{ij} - x_{oj})^2 + (y_{ij} - y_{oj})^2 + f_j^{\,2}}} \tag{79}$$

Unknown Camera Position and Attitude

The first attempt to linearize the colinearity equations assumes that the points' locations and the positions and attitudes of the cameras are unknown, as in strategy 2. In this case, the known and unknown values are as follows:

$$known: x_{ij}, y_{ij}, x_{oj}, y_{oj}, f_j,$$
$$unknown: X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, \alpha_{ij}, C_{rsj}$$

Manipulating the colinearity equations from Chapter III to separate the known and unknown quantities produces the following equation:

$$\left[C_j\right]^T \alpha_{ij} \begin{Bmatrix} x_{ij} - x_{oj} \\ y_{ij} - y_{oj} \\ f_j \end{Bmatrix} - \begin{Bmatrix} X_i \\ Y_i \\ Z_i \end{Bmatrix} + \begin{Bmatrix} X_{Cj} \\ Y_{Cj} \\ Z_{Cj} \end{Bmatrix} = 0 \tag{80}$$

Before putting this into matrix form, it is realized that there are quadratically more unknown values than known values, and as the number of points and/or number of cameras increase, the system becomes more underdetermined. Therefore, this strategy cannot be solved using this linearization technique.

Absolute Camera Position and Attitude

The next linearization method assumes that the points are unknown, but all camera information is known, as in strategy 1. In this case, the known values and unknown values are as follows:

$$known: x_{ij}, y_{ij}, x_{oj}, y_{oj}, f_j, X_{Cj}, Y_{Cj}, Z_{Cj}, C_{rsj}$$
$$unknown: X_i, Y_i, Z_i, \alpha_{ij}$$

The colinearity equation can be rearranged to have a known value multiplied by an unknown value.

$$\alpha_{ij}\begin{Bmatrix} x_{ij}-x_{oj} \\ y_{ij}-y_{oj} \\ f_j \end{Bmatrix} - [C_j]\begin{Bmatrix} X_i - X_{Cj} \\ Y_i - Y_{Cj} \\ Z_i - Z_{Cj} \end{Bmatrix} = 0 \tag{81}$$

At this point, the equation would be separated into a form where a matrix of known values is multiplied by a vector of unknown values, to find the null space of the known matrix. This cannot be done however, because (X_{Cj}, Y_{Cj}, Z_{Cj}) have no unknown variable by which the vector containing the camera positions are multiplied. To overcome this issue, there are two methods to solve this strategy: assume the camera attitude is unknown or assume the camera orientation is unknown.

Camera Attitude Unknown

Assume the points and camera attitude are unknown. In this case, the known and unknown values are as follows:

$$known: x_{ij}, y_{ij}, x_{oj}, y_{oj}, f_j, X_{Cj}, Y_{Cj}, Z_{Cj}$$
$$unknown: X_i, Y_i, Z_i, \alpha_{ij}, C_{rsj}$$

$$[C_{rsj}]^T\begin{Bmatrix} x_{ij}-x_{oj} \\ y_{ij}-y_{oj} \\ f_j \end{Bmatrix} - \frac{1}{\alpha_{ij}}\begin{Bmatrix} X_i - X_{Cj} \\ Y_i - Y_{Cj} \\ Z_i - Z_{Cj} \end{Bmatrix} = 0 \tag{82}$$

Using this equation, the number of unknowns greatly outweighs the number of equations, because a new variable is introduced when α_{ij} is multiplied by (X,Y, Z), so the unknowns are now $\alpha_{ij}, a_{ij}, b_{ij}, c_{ij}$ and the 9 elements of C, multiplied by the number of

cameras. For a 2 camera system, the number of unknowns are 8*p+18, and the number of equations are only 6*p. This problem is underdetermined.

Camera Position Unknown

Assume the entire second bracketed quantity in Eq. 81 is an unknown value, but the attitude information is still known:

$$known: x_{ij}, y_{ij}, x_{oj}, y_{oj}, f_j, C_{rsj}$$
$$unknown: X_i, Y_i, Z_i, X_{Cj}, Y_{Cj}, Z_{Cj}, \alpha_{ij}$$

To ensure the system can be solved, first solve for a 2 camera system. There are 5*p+6 unknown values and 6*p equations, so this problem can be solved, as long as p≥6. The following shows Eq. (81) as a matrix of known values multiplied by a vector of unknown values for a 2 camera system:

$$\left[\begin{array}{ccccc|ccc|ccc|c}
x_{11}-x_{01} & & & & & & & & & & & \\
y_{11}-y_{01} & 0 & 0 & 0 & & -C_{111} & -C_{121} & -C_{131} & C_{111} & C_{121} & C_{131} & \\
-f_1 & & & & & -C_{211} & -C_{221} & -C_{231} & C_{211} & C_{221} & C_{231} & \\
 & x_{12}-x_{01} & & & & -C_{311} & -C_{321} & -C_{331} & C_{311} & C_{321} & C_{331} & \\
0 & y_{12}-y_{01} & 0 & 0 & & 0 & & \ddots & \vdots & & 0 & \\
 & -f_1 & & & & -C_{112} & -C_{122} & -C_{132} & & & C_{112} \; C_{122} \; C_{132} & \\
0 & 0 & \ddots & 0 & & -C_{212} & -C_{222} & -C_{232} & 0 & & C_{212} \; C_{222} \; C_{232} & \\
 & & & x_{2p}-x_{02} & & -C_{312} & -C_{322} & -C_{332} & & & C_{312} \; C_{322} \; C_{332} & \\
0 & 0 & 0 & y_{2p}-y_{02} & & 0 & & \ddots & 0 & & \vdots & \\
 & & & -f_2 & &
\end{array}\right]
\left[\begin{array}{c}
\alpha_{11} \\ \alpha_{12} \\ \vdots \\ \alpha_{2p} \\ \hline X_1 \\ Y_1 \\ Z_1 \\ \vdots \\ X_p \\ Y_p \\ Z_p \\ \hline X_{C1} \\ Y_{C1} \\ Z_{C1} \\ X_{C2} \\ Y_{C2} \\ Z_{C2}
\end{array}\right] = 0$$

(83)

For p=6, using the null and rank functions in Matlab, the matrix is rank deficient.[8] There is no vector that exactly satisfies the null space condition because one of the rows or columns in the matrix is exactly the same as another row or column.[9] For a 2 camera system of 6 points, this is not a useful linearization technique.

Linear Least Squares Optimization

For the first strategy, the equations from the last section can be put into a linear least squares loop to determine the (X_i, Y_i, Z_i). Linear least squares estimation is preferred to the nonlinear least squares estimation because the linear estimation has a shorter run-time and is less complex than nonlinear estimation. Also, the linear method can calculate the entire array of cameras, producing all of the points very quickly. Again, the known and unknown values are as follows:

$$known: x_{ij}, y_{ij}, x_{oj}, y_{oj}, f_j, X_{Cj}, Y_{Cj}, Z_{Cj}, C_{rsj}$$
$$unknown: X_i, Y_i, Z_i, \alpha_{ij}$$

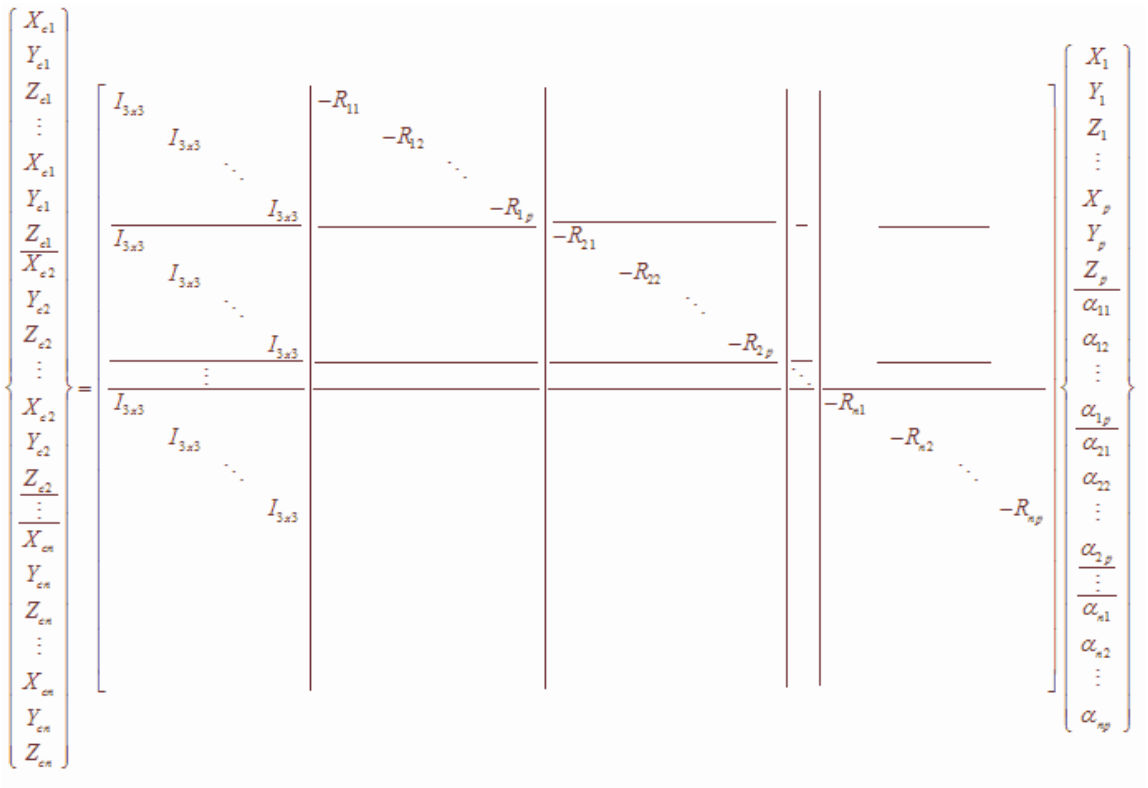
The colinearity equations are rearranged to solve for the known values (X_{cj}, Y_{cj}, Z_{cj}) :

$$\begin{Bmatrix} X_i \\ Y_i \\ Z_i \end{Bmatrix} - [C_j]^T \begin{Bmatrix} x_{ij} - x_{oj} \\ y_{ij} - y_{oj} \\ f_j \end{Bmatrix} \alpha_{ij} = \begin{Bmatrix} X_{Cj} \\ Y_{Cj} \\ Z_{Cj} \end{Bmatrix} \tag{84}$$

A new variable R_{ij} is created to sum up the known quantities for each point of each camera:

$$R_{ij} \equiv [C_j]^T \begin{Bmatrix} x_{ij} - x_{oj} \\ y_{ij} - y_{oj} \\ f_j \end{Bmatrix} \tag{85}$$

Eq. (85) is separated into a matrix of known values multiplied by a vector of unknown values, which is now set equal to a vector of known values.



(86)

Using the linearized approximation method with a linear least squares program instead of the nonlinear least squares program has increased the accuracy and shortened the time required for the calculation. These surfaces are modeled with a noise value of 10^{-3}m, which is 1000 times larger than noise value used in the previous chapters. The actual surface (Figure 35) is created using 10 cameras and a matrix of 7x7 projected points. The surface is located 3.7m from the camera lens, is 100m long and 7m deep. Figure 36 shows the model that is computed with the nonlinear least squares program. Figure 37 contains the model using the new linear least squares program.

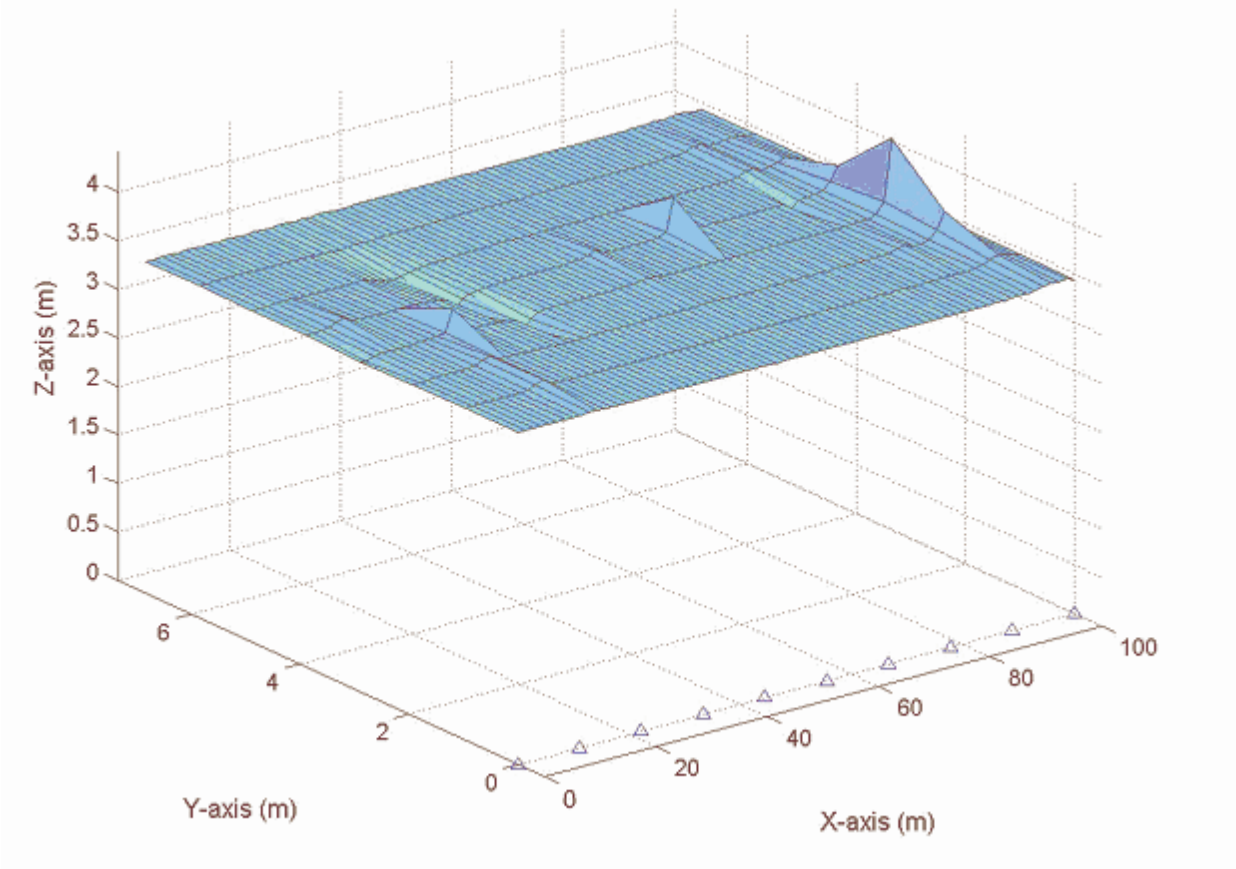


Fig. 35 Antennae Surface.

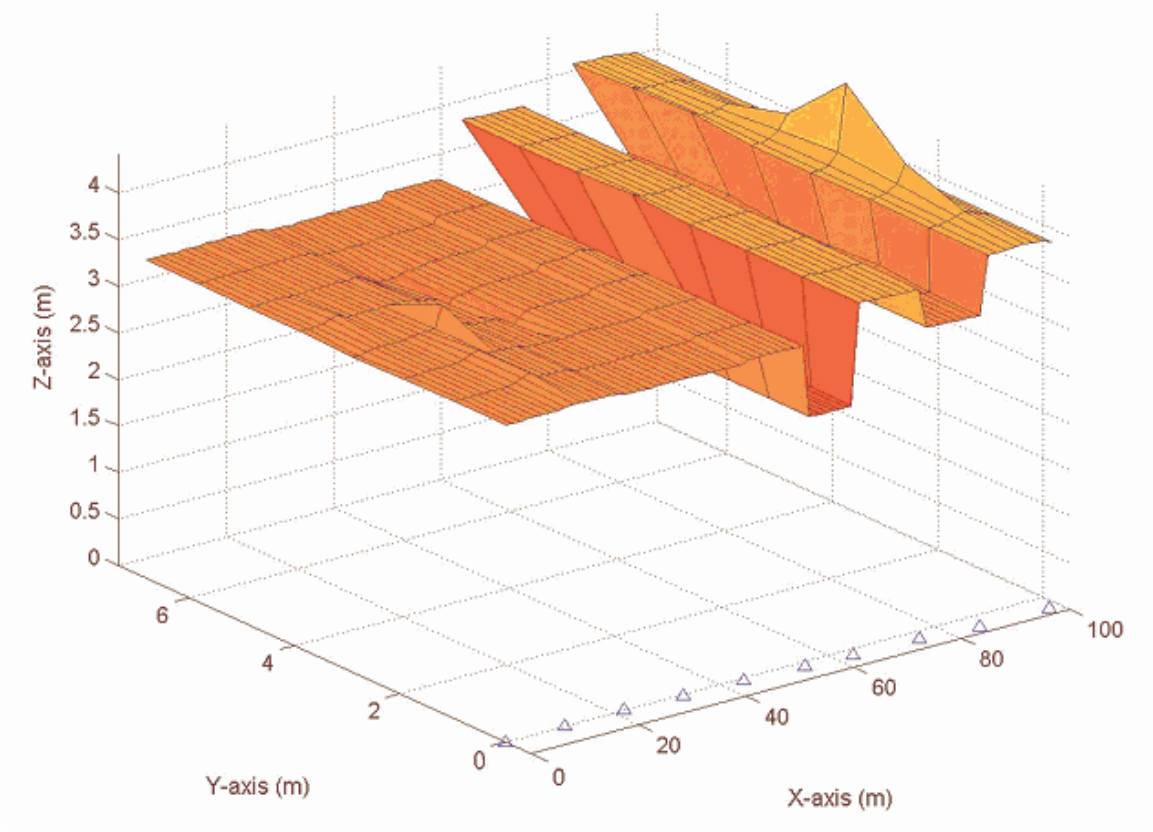


Fig. 36 Nonlinear Least Squares Approximation.

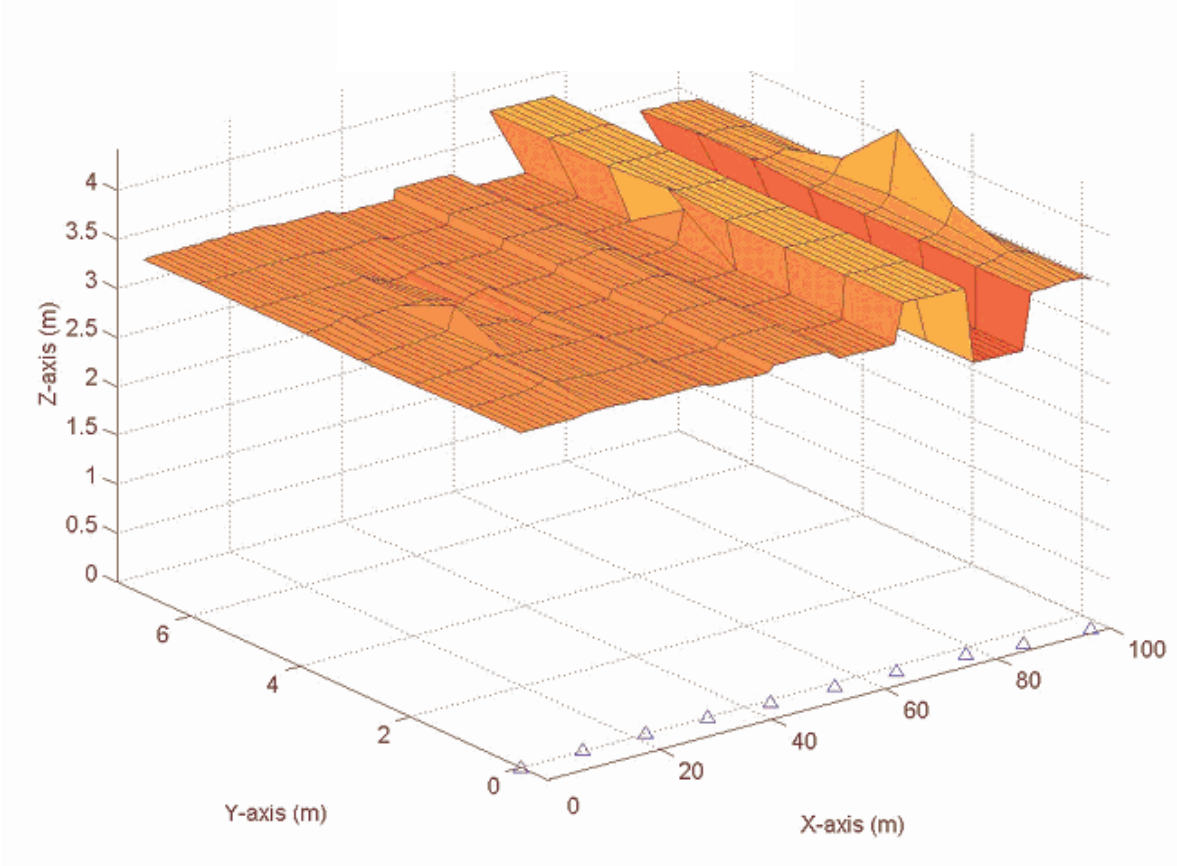


Fig. 37 Linear Least Squares Approximation.

While a linearized closed form solution of the colinearity equations has not been found, a linearization method that improves accuracy and speed of the first strategy has been established. The main hindrance in creating a closed form solution for the colinearity equations occurs as a result of the rotation matrix. There are 9 elements that are unknown when the attitude is unknown. Using a parameterization for the rotation matrix would not be any less complex because all parameterizations involve additional nonlinear equations to create the rotation matrix. With this in mind, a no attitude approach to linearizing the colinearity equations is explored in Appendix C, but it is found that for the number of points required to accurately model a surface (on the order

of 10^5), the complexity increases dramatically to the point where the linearization involves more computation than a nonlinear Gaussian least squares program.

The only method that truly linearizes the multiple stereo pair problem is to assume that only the points are unknown and the cameras attitudes and locations are known. While this assumption makes it necessary to know the camera information, it requires only a linear least squares program, reduced from the previous, more complex nonlinear least squares program, and it increases the accuracy significantly.

APPENDIX B

GLOBAL METROLOGY SENSORS

One solution to finding the locations of the cameras for strategy 3 is to use global metrology, which would give approximate a priori knowledge of the locations of the cameras locations and attitudes. A better estimate of the locations of the cameras should dramatically reduce the errors and approximate the surface more accurately. Global metrology uses a laser flashlight above the inboard-most camera, a laser ranging device to that camera, and retro reflectors attached at three corners of each camera. Figure 38 shows the general configuration, and Figure 39 shows the positioning of the retro reflectors on each camera.

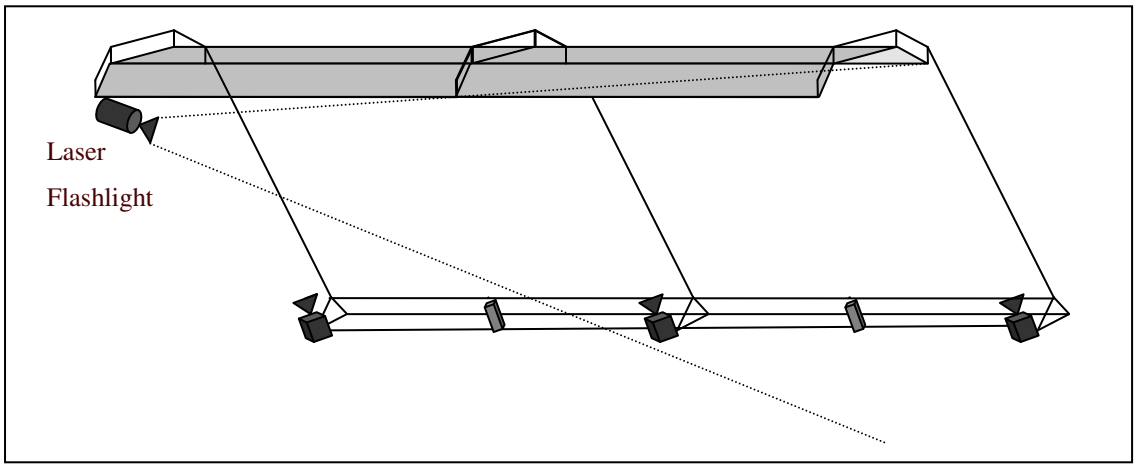


Fig. 38 Assembly of Global Metrology Sensors.

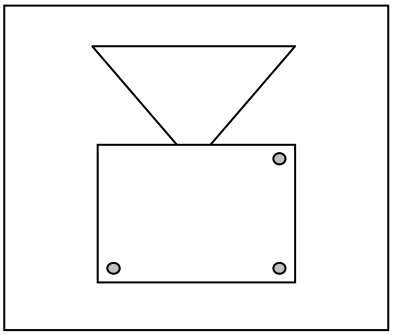


Fig. 39 Retro-Reflectors Positioning on Camera.

Using the laser ranging device to send a signal and receive feedback to each corner of the camera, the range, position, and attitude can be found by sequentially 'opening' each retro reflector. The laser flashlight that illuminates the reflective material can also be used to improve the visibility of the laser dots on the surface of the antennae, and thus allow them to appear in the images more clearly. The retro-reflectors are currently being developed by the Navy Research Laboratory, and referred to as Modulating Retro-Reflector (MRR). [10] The MRR includes an optical retro-reflector and a shutter that modulates the laser beam from the center camera to every other camera, and is shown in Figure 40.

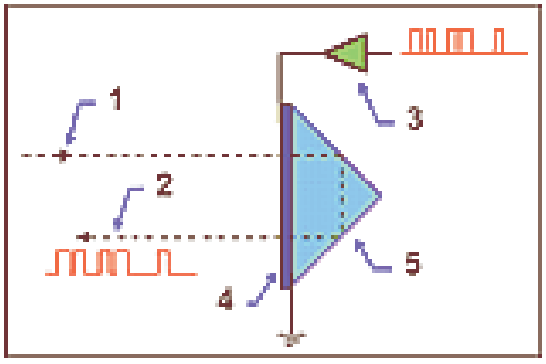


Fig. 40 Modulating Retro-Reflector.

A constraint with global metrology is that the density of the light must be known and included in the design to accurately measure the range.[11] Also, geometrical

constraints arise to measure all of the cameras. The ideal geometric arrangement is shown in Figure 41, so all retro reflectors are visible and in line.

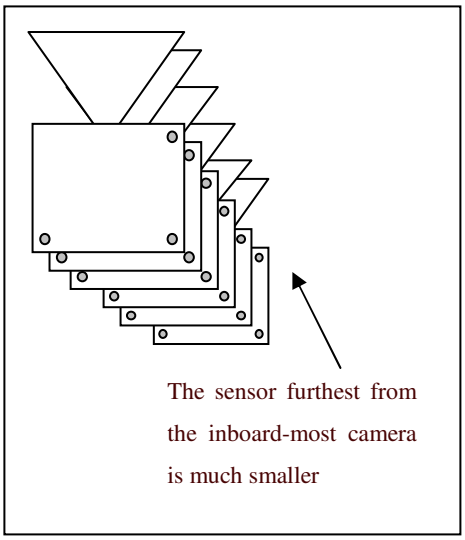


Fig. 41 Ideal Geometric Alignment of Cameras.

Using global metrology and MRR's produces a better estimate of the cameras' locations and dramatically reduces the errors. Thus, the surface is approximated more accurately with the use of the additional sensors, as seen in strategy 3.

APPENDIX C

NO ATTITUDE APPROACH

The largest complexity associated with Appendix A was the attitude matrix. A no attitude approach can decrease the number of unknowns so the points' locations can be solved for more easily. This method[7] can be adapted for a stereo pair system. This method uses different notation than the rest of the thesis, where $1 \le m \le 2$ and $1 \le k \le p$. The camera is located at:

$$r_{cm} = \begin{bmatrix} X_{cm} \\ Y_{cm} \\ Z_{cm} \end{bmatrix} \tag{87}$$

The points are located at:

$$r_k = \begin{bmatrix} X_k \\ Y_k \\ Z_k \end{bmatrix} \tag{88}$$

:
$$m_{km} = \sqrt{(X_k - X_{cm})^2 + (Y_k - Y_{cm})^2 + (Z_k - Z_{cm})^2} \tag{89}$$

$$\hat{\underline{b}}_{km} = \frac{1}{\sqrt{(x_k - x_{om})^2 + (y_k - y_{om})^2 + f_m^{\,2}}} \begin{bmatrix} x_k - x_{om} \\ y_k - y_{om} \\ f_m \end{bmatrix} \tag{90}$$

Known Values: $\hat{\underline{b}}_{km}$

Unknown Values: r_{cm}, r_k, m_{km}

Using the matrix form of the colinearity equation:

$$C_m \frac{1}{m_{km}}(r_k - r_m) = \hat{\underline{b}}_{km} \tag{91}$$

For the first camera (m=1), and the points on the antennae (i, j, k):

$$C_1 \frac{1}{m_{i1}}(r_i - r_{c1}) = \hat{\underline{b}}_{1i} \tag{92}$$

$$C_1 \frac{1}{m_{j1}}(r_{1j} - r_{c1}) = \hat{\underline{b}}_{1j} \tag{93}$$

$$C_1 \frac{1}{m_{k1}}(r_k - r_{c1}) = \hat{\underline{b}}_{1k} \tag{94}$$

Combine these equations into the form:

$$C_1 \begin{bmatrix} r_i - r_j \\ r_j - r_k \\ r_i - r_k \end{bmatrix} = \begin{bmatrix} m_{i1}\hat{\underline{b}}_{i1} - m_{j1}\hat{\underline{b}}_{j1} \\ m_{j1}\hat{\underline{b}}_{j1} - m_{k1}\hat{\underline{b}}_{k1} \\ m_{i1}\hat{\underline{b}}_{i1} - m_{k1}\hat{\underline{b}}_{k1} \end{bmatrix} \tag{95}$$

This can be reduced by renaming the bracketed values:

$$C_1 R = B_1 \tag{96}$$

$$R^T C_1^T = B_1^T \tag{97}$$

These can be combined to remove C from the equation:

$$R^T R = B_1^T B_1 \tag{98}$$

$$R = \begin{bmatrix} X_i - X_j & Y_i - Y_j & Z_i - Z_j \\ X_j - X_k & Y_j - Y_k & Z_j - Z_k \\ X_i - X_k & Y_i - Y_k & Z_i - Z_k \end{bmatrix} \tag{99}$$

$$B_1 = \begin{bmatrix} \alpha_{i1}(x_{c1} - x_i) - \alpha_{j1}(x_{c1} - x_j) & \alpha_{i1}(y_{c1} - y_i) - \alpha_{j1}(y_{c1} - y_j) & f_1(\alpha_{j1} - \alpha_{i1}) \\ \alpha_{j1}(x_{c1} - x_j) - \alpha_{k1}(x_{c1} - x_k) & \alpha_{j1}(y_{c1} - y_j) - \alpha_{k1}(y_{c1} - y_k) & f_1(\alpha_{k1} - \alpha_{j1}) \\ \alpha_{i1}(x_{c1} - x_i) - \alpha_{k1}(x_{c1} - x_k) & \alpha_{i1}(y_{c1} - y_i) - \alpha_{k1}(y_{c1} - y_k) & f_1(\alpha_{k1} - \alpha_{i1}) \end{bmatrix} \tag{100}$$

Multiply the matrices by their transpose, and equate the elements to solve for (X, Y, Z). This construction is to find only three points' location. Considering the application of this is to generate an entire surface, thousands of points are actually needed. To generate an array of thousands of points using this no attitude method would require much more computation and complexity than the current program, so it is not utilized, but is included because it is a method that was tried and failed for this application.

VITA

Elizabeth Osgood received her Bachelor of Science degree in Aerospace Engineering from Embry-Riddle Aeronautical University in Prescott, AZ in 2004. She began her Master's degree work at Texas A&M in Aerospace Engineering in fall of 2004. Her research interests are attitude determination and control, with focus on vision-based sensors. She also enjoys micro-satellite design.

Elizabeth's permanent address is: 9618 Antoine Forest Dr., San Antonio, TX 78254. Her email is frekles83@yahoo.com.